WHAT IS CLAIMED IS:

1. A latency-independent interface between first and second hardware components, comprising:

a serial control data circuit that transmits a serial control data signal; and

a data circuit that transmits or receives data under the control of the serial data gate signal,

wherein the serial control data signal comprises information as to whether the data is one of split and non-split.

2. The latency-independent interface of claim 1, wherein the serial control data signal comprises information that the data is one of first split, continue split and last split.

3. The latency-independent interface of claim 1, wherein the serial control data signal comprises an amount of the data to be written during a write operation.

4. The latency-independent interface of claim 1, wherein the serial control data signal comprises an amount of the data to be read during a read operation.

5. The latency-independent interface of claim 1, wherein the serial control data signal comprises a codeword size of a current sector.

6. The latency-independent interface of claim 1, wherein the serial control data signal comprises a information if a succeeding serial control data is a continuation of a current serial control data.

7. The latency-independent interface of claim 1, wherein the serial control data signal comprises

during a write operation information as to a start of a sync mark and a start of write padding data, and

during a read operation information that a sync mark was detected.

8. The latency-independent interface of claim 1, further comprising a ready transceiver that transmits or receives a bi-directional ready signal.

9. The latency-independent interface of claim 1, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

10. The latency-independent interface of claim 1, further comprising a sync mark transceiver that transmits or receives sync mark information.

11. The latency-independent interface of claim 10, wherein during a write operation a first assertion of the sync mark information indicates a start of sync mark insertion and a second assertion of the sync mark information indicates a start of writing of padding data.

12. A latency-independent interface between first and second hardware components, comprising:

a serial control data circuit that transmits a serial control data signal;

a data circuit that transmits or receives data under the control of the serial control data signal; and

a sync mark transceiver that transmits or receives sync mark information,

wherein during a write operation a first assertion by the first hardware component of the sync mark information indicates a start of sync mark insertion and a second assertion by the first hardware component of the sync mark information indicates a start of writing of padding data, and

during a read operation by the second hardware component information that a sync mark was detected.

13. The latency-independent interface of claim 12, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

14. A latency-independent interface between first and second hardware components, comprising:

a serial control data circuit that transmits a serial control data signal;

a data circuit that transmits or receives data under the control of the serial data gate signal; and

a ready transceiver that transmits or receives a ready signal;

during a write operation the ready signal indicates the second hardware component is ready to receive data from the first hard component; and

during a read operation the ready signal indicates the first hardware component is ready to receive data from the second hard.

15. A latency-independent interface between first and second hardware components, comprising:

serial control transmitting means for transmitting a serial control data signal; and

data transceiver means for transmitting or receiving data under the control of the serial control data signal,

wherein the serial control data signal comprises information as to whether the data is one of split and non-split.

16. The latency-independent interface of claim 15, wherein the serial control data signal comprises information that the data is one of first split, continue split and last split.

17. The latency-independent interface of claim 15, wherein the serial control data signal comprises an amount of the data to be written during a write operation.

18. The latency-independent interface of claim 15, wherein the serial control data signal comprises an amount of the data to be read during a read operation.

19. The latency-independent interface of claim 15, wherein the serial control data signal comprises a codeword size of a current sector.

20. The latency-independent interface of claim 15, wherein the serial control data signal comprises a information if a succeeding serial control data is a continuation of a current serial control data.

21. The latency-independent interface of claim 15, wherein the serial control data signal comprises

during a write operation information as to a start of a sync mark and a start of write padding data, and

during a read operation information that a sync mark was detected.

22. The latency-independent interface of claim 15, further comprising a ready transceiver means for transmitting or receiving a bi-directional ready signal.

23. The latency-independent interface of claim 15, wherein the first hardware component comprises disk controller means and the second hardware component comprises read channel means.

24. The latency-independent interface of claim 15, further comprising a sync mark transceiver means for transmitting or receiving sync mark information.

25. The latency-independent interface of claim 24, wherein during a write operation a first assertion of the sync mark information indicates a start of sync mark insertion and a second assertion of the sync mark information indicates a start of writing of padding data.

26. A latency-independent interface between first and second hardware components, comprising:

serial control transmitting means for transmitting a serial control data signal;

data transceiver means for transmitting or receiving data under the control of the serial control data signal; and

sync mark transceiver means for transmitting or receiving sync mark information,

wherein during a write operation a first assertion by the first hardware component of the sync mark information indicates a start of sync mark insertion and a second assertion by the first hardware component of the sync mark information indicates a start of writing of padding data, and

during a read operation by the second hardware component information that a sync mark was detected.

27. The latency-independent interface of claim 26, wherein the first hardware component comprises disk controller means and the second hardware component comprises read channel means.

28. A latency-independent interface between first and second hardware components, comprising:

serial control transmitting means for transmitting a serial control data signal;

data transceiver means for transmitting or receiving data under the control of the serial control data signal; and

ready transceiver means for transmitting or receiving a ready signal;

during a write operation the ready signal indicates the second hardware component is ready to receive data from the first hard component; and

during a read operation the ready signal indicates the first hardware component is ready to receive data from the second hard component.

29. The latency-independent interface of claim 28, wherein the first hardware component comprises disk controller means and the second hardware component comprises read channel means.

30. The latency-independent interface of claim 14, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

31. A method of transmitting and receiving signals between first and second hardware components, comprising the steps of:

transmitting a serial control data signal; and

transmitting or receiving data under the control of the serial control data signal,

wherein the serial control data signal comprises information as to whether the data is one of split and non-split.

32. The method of claim 31, wherein the serial control data signal comprises information that the data is one of first split, continue split and last split.

33. The method of claim 31, wherein the serial control data signal comprises an amount of the data to be written during a write operation.

34. The method of claim 31, wherein the serial control data signal comprises an amount of the data to be read during a read operation.

35. The method of claim 31, wherein the serial control data signal comprises a codeword size of a current sector.

36. The method of claim 31, wherein the serial control data signal comprises information if a succeeding serial control data is a continuation of a current serial control data.

37. The method of claim 31, wherein the serial control data signal comprises:

during a write operation information as to a start of a sync mark and a start of write padding data, and

during a read operation information that a sync mark was detected.

38. The method of claim 31, further comprising the step of transmitting or receiving a bi-directional ready signal.

39. The method of claim 31, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

40. The method of claim 31, further comprising the step of transmitting or receiving sync mark information.

41. The method of claim 40, wherein during a write operation a first assertion of the sync mark information indicates a start of sync mark insertion and a second assertion of the sync mark information indicates a start of writing of padding data.

42. A method of transmitting and receiving signals between first and second hardware components, comprising the steps of:

transmitting a serial control data signal;

transmitting or receiving data under the control of the serial control data signal; and

transmitting or receiving sync mark information,

wherein during a write operation a first assertion by the first hardware component of the sync mark information indicates a start of sync mark insertion and a second assertion by the first hardware component of the sync mark information indicates a start of writing of padding data, and

during a read operation by the second hardware component information that a sync mark was detected.

43. The method of claim 42, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

44. A method of transmitting and receiving signals between first and second hardware components, comprising the steps of:

transmitting a serial control data signal;

transmitting or receiving data under the control of the serial control data signal; and

transmitting or receiving a ready signal;

during a write operation the ready signal indicates the second hardware component is ready to receive data from the first hard component; and

during a read operation the ready signal indicates the first hardware component is ready to receive data from the second hard component.

45. The method of claim 44, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

46. A latency-independent interface between first and second hardware components, comprising:

a serial control data circuit that receives a serial control data signal; and

a data circuit that transmits or receives data under the control of the serial data gate signal,

wherein the serial control data signal comprises information as to whether the data is one of split and non-split.

47. The latency-independent interface of claim 46, wherein the serial control data signal comprises information that the data is one of first split, continue split and last split.

48. The latency-independent interface of claim 46, wherein the serial control data signal comprises an amount of the data to be written during a write operation.

49. The latency-independent interface of claim 46, wherein the serial control data signal comprises an amount of the data to be read during a read operation.

50. The latency-independent interface of claim 46, wherein the serial control data signal comprises a codeword size of a current sector.

51. The latency-independent interface of claim 46, wherein the serial control data signal comprises information if a succeeding serial control data is a continuation of a current serial control data.

52. The latency-independent interface of claim 46, wherein the serial control data signal comprises:

during a write operation information as to a start of a sync mark and a start of write padding data, and

during a read operation information that a sync mark was detected.

53. The latency-independent interface of claim 46, further comprising a ready transceiver that transmits or receives a bi-directional ready signal.

54. The latency-independent interface of claim 46, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

55. The latency-independent interface of claim 46, further comprising a sync mark transceiver that transmits or receives sync mark information.

56. The latency-independent interface of claim 55, wherein during a write operation a first assertion of the sync mark information indicates a start of sync mark insertion and a second assertion of the sync mark information indicates a start of writing of padding data.

57. A latency-independent interface between first and second hardware components, comprising:

a serial control data circuit that receives a serial control data signal;

a data circuit that transmits or receives data under the control of the serial control data signal; and

a sync mark transceiver that transmits or receives sync mark information,

wherein during a write operation a first assertion by the first hardware component of the sync mark information indicates a start of sync mark insertion and a second assertion by the first hardware component of the sync mark information indicates a start of writing of padding data, and

during a read operation by the second hardware component information that a sync mark was detected.

58. The latency-independent interface of claim 57, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

59. A latency-independent interface between first and second hardware components, comprising:

a serial control data circuit that receives a serial control data signal;

a data circuit that transmits or receives data under the control of the serial data gate signal; and

a ready transceiver that transmits or receives a ready signal;

during a write operation the ready signal indicates the second hardware component is ready to receive data from the first hard component; and

during a read operation the ready signal indicates the first hardware component is ready to receive data from the second hard.

60. A latency-independent interface between first and second hardware components, comprising:

serial control receiving means for receiving a serial control data signal; and

data transceiver means for transmitting or receiving data under the control of the serial control data signal,

wherein the serial control data signal comprises information as to whether the data is one of split and non-split.

61. The latency-independent interface of claim 60, wherein the serial control data signal comprises information that the data is one of first split, continue split and last split.

62. The latency-independent interface of claim 60, wherein the serial control data signal comprises an amount of the data to be written during a write operation.

63. The latency-independent interface of claim 60, wherein the serial control data signal comprises an amount of the data to be read during a read operation.

64. The latency-independent interface of claim 60, wherein the serial control data signal comprises a codeword size of a current sector.

65. The latency-independent interface of claim 60, wherein the serial control data signal comprises information if a succeeding serial control data is a continuation of a current serial control data.

66. The latency-independent interface of claim 60, wherein the serial control data signal comprises

during a write operation information as to a start of a sync mark and a start of write padding data, and

during a read operation information that a sync mark was detected.

67. The latency-independent interface of claim 60, further comprising a ready transceiver means for transmitting or receiving a bi-directional ready signal.

68. The latency-independent interface of claim 60, wherein the first hardware component comprises disk controller means and the second hardware component comprises read channel means.

69. The latency-independent interface of claim 60, further comprising a sync mark transceiver means for transmitting or receiving sync mark information.

70. The latency-independent interface of claim 69, wherein during a write operation a first assertion of the sync mark information indicates a start of sync mark insertion

and a second assertion of the sync mark information indicates a start of writing of padding data.

71. A latency-independent interface between first and second hardware components, comprising:

serial control receiving means for receiving a serial control data signal;

data transceiver means for transmitting or receiving data under the control of the serial control data signal; and

sync mark transceiver means for transmitting or receiving sync mark information,

wherein during a write operation a first assertion by the first hardware component of the sync mark information indicates a start of sync mark insertion and a second assertion by the first hardware component of the sync mark information indicates a start of writing of padding data, and

during a read operation by the second hardware component information that a sync mark was detected.

72.The latency-independent interface of claim 71, wherein the first hardware component comprises disk controller means and the second hardware component comprises read channel means.

73. A latency-independent interface between first and second hardware components, comprising:

serial control receiving means for receiving a serial control data signal;

data transceiver means for transmitting or receiving data under the control of the serial control data signal; and

ready transceiver means for transmitting or receiving a ready signal;

during a write operation the ready signal indicates the second hardware component is ready to receive data from the first hard component; and

during a read operation the ready signal indicates the first hardware component is ready to receive data from the second hard component.

74. The latency-independent interface of claim 73, wherein the first hardware component comprises disk controller means and the second hardware component comprises read channel means.

75. The latency-independent interface of claim 59, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

76. A method of transmitting and receiving signals between first and second hardware components, comprising the steps of:

receiving a serial control data signal; and

transmitting or receiving data under the control of the serial control data signal,

wherein the serial control data signal comprises information as to whether the data is one of split and non-split.

77. The method of claim 76, wherein the serial control data signal comprises information that the data is one of first split, continue split and last split.

78. The method of claim 76, wherein the serial control data signal comprises an amount of the data to be written during a write operation.

79. The method of claim 76, wherein the serial control data signal comprises an amount of the data to be read during a read operation.

80. The method of claim 76, wherein the serial control data signal comprises a codeword size of a current sector.

81. The method of claim 76, wherein the serial control data signal comprises information if a succeeding serial control data is a continuation of a current serial control data.

82. The method of claim 76, wherein the serial control data signal comprises

during a write operation information as to a start of a sync mark and a start of write padding data, and

during a read operation information that a sync mark was detected.

83. The method of claim 76, further comprising the step of transmitting or receiving a bi-directional ready signal.

84. The method of claim 76, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

85. The method of claim 76, further comprising the step of transmitting or receiving sync mark information.

86. The method of claim 85, wherein during a write operation a first assertion of the sync mark information indicates a start of sync mark insertion and a second assertion of the sync mark information indicates a start of writing of padding data.

87. A method of transmitting and receiving signals between first and second hardware components, comprising the steps of:

receiving a serial control data signal;

transmitting or receiving data under the control of the serial control data signal; and

transmitting or receiving sync mark information,

wherein during a write operation a first assertion by the first hardware component of the sync mark information indicates a start of sync mark insertion and a second assertion by the first hardware component of the sync mark information indicates a start of writing of padding data, and

during a read operation by the second hardware component information that a sync mark was detected.

88. The method of claim 86, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

89. A method of transmitting and receiving signals between first and second hardware components, comprising the steps of:

receiving a serial control data signal;

transmitting or receiving data under the control of the serial control data signal; and

transmitting or receiving a ready signal;

during a write operation the ready signal indicates the second hardware component is ready to receive data from the first hard component; and

during a read operation the ready signal indicates the first hardware component is ready to receive data from the second hard component.

90. The method of claim 89, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

91. A computer program for transmitting and receiving signals between first and second hardware components, comprising the steps of:

receiving a serial control data signal; and

transmitting or receiving data under the control of the serial control data signal,

wherein the serial control data signal comprises information as to whether the data is one of split and non-split.

92. The computer program of claim 91, wherein the serial control data signal comprises information that the data is one of first split, continue split and last split.

93. The computer program of claim 91, wherein the serial control data signal comprises an amount of the data to be written during a write operation.

94. The computer program of claim 91, wherein the serial control data signal comprises an amount of the data to be read during a read operation.

95. The computer program of claim 91, wherein the serial control data signal comprises a codeword size of a current sector.

96. The computer program of claim 91, wherein the serial control data signal comprises information if a succeeding serial control data is a continuation of a current serial control data.

97. The computer program of claim 91, wherein the serial control data signal comprises:

during a write operation information as to a start of a sync mark and a start of write padding data, and

during a read operation information that a sync mark was detected.

98. The computer program of claim 91, further comprising the step of transmitting or receiving a bi-directional ready signal.

99. The computer program of claim 91, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

100. The computer program of claim 91, further comprising the step of transmitting or receiving sync mark information.

101. The computer program of claim 100, wherein during a write operation a first assertion of the sync mark information indicates a start of sync mark insertion and a second assertion of the sync mark information indicates a start of writing of padding data.

102. A computer program for transmitting and receiving signals between first and second hardware components, comprising the steps of:

receiving a serial control data signal;

transmitting or receiving data under the control of the serial control data signal; and

transmitting or receiving sync mark information,

wherein during a write operation a first assertion by the first hardware component of the sync mark information indicates a start of sync mark insertion and a second assertion by the first hardware component of the sync mark information indicates a start of writing of padding data, and

during a read operation by the second hardware component information that a sync mark was detected.

103. The computer program of claim 102, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

104. A computer program for transmitting and receiving signals between first and second hardware components, comprising the steps of:

receiving a serial control data signal;

transmitting or receiving data under the control of the serial control data signal; and

transmitting or receiving a ready signal;

during a write operation the ready signal indicates the second hardware component is ready to receive data from the first hard component; and

during a read operation the ready signal indicates the first hardware component is ready to receive data from the second hard component.

105. The computer program of claim 104, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

106. A computer program for transmitting and receiving signals between first and second hardware components, comprising the steps of:

transmitting a serial control data signal; and

transmitting or receiving data under the control of the serial control data signal,

wherein the serial control data signal comprises information as to whether the data is one of split and non-split.

107. The computer program of claim 106, wherein the serial control data signal comprises information that the data is one of first split, continue split and last split.

108. The computer program of claim 106, wherein the serial control data signal comprises an amount of the data to be written during a write operation.

109. The computer program of claim 106, wherein the serial control data signal comprises an amount of the data to be read during a read operation.

110. The computer program of claim 106, wherein the serial control data signal comprises a codeword size of a current sector.

111. The computer program of claim 106, wherein the serial control data signal comprises a information if a succeeding serial control data is a continuation of a current serial control data.

112. The computer program of claim 106, wherein the serial control data signal comprises:

during a write operation information as to a start of a sync mark and a start of write padding data, and

during a read operation information that a sync mark was detected.

113. The computer program of claim 106, further comprising the step of transmitting or receiving a bi-directional ready signal.

114. The computer program of claim 106, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

115. The computer program of claim 106, further comprising the step of transmitting or receiving sync mark information.

116. The computer program of claim 115, wherein during a write operation a first assertion of the sync mark information indicates a start of sync mark insertion and a second assertion of the sync mark information indicates a start of writing of padding data.

117. A computer program for transmitting and receiving signals between first and second hardware components, comprising the steps of:

transmitting a serial control data signal;

transmitting or receiving data under the control of the serial control data signal; and

transmitting or receiving sync mark information,

wherein during a write operation a first assertion by the first hardware component of the sync mark information indicates a start of sync mark insertion and a second assertion by the first hardware component of the sync mark information indicates a start of writing of padding data, and

during a read operation by the second hardware component information that a sync mark was detected.

118. The computer program of claim 117, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

119. A computer program for transmitting and receiving signals between first and second hardware components, comprising the steps of:

transmitting a serial control data signal;

transmitting or receiving data under the control of the serial control data signal; and

transmitting or receiving a ready signal;

during a write operation the ready signal indicates the second hardware component is ready to receive data from the first hard component; and

during a read operation the ready signal indicates the first hardware component is ready to receive data from the second hard component.

120. The computer program of claim 119, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

121. A data transmission system comprising:

a first component comprising:

a serial control transmitter circuit that transmits a serial control data signal; and

a first data transceiver circuit that transmits or receives data under the control of the serial control data signal,

a second component comprising:

a serial control receiver circuit that receives the serial control data signal; and

a second data transceiver circuit that transmits or receives the data under the control of the serial control data signal,

wherein the serial control data signal comprises information as to whether the data is one of split and non-split.

122. The latency-independent interface of claim 121, wherein the serial control data signal comprises information that the data is one of first split, continue split and last split.

123. The latency-independent interface of claim 121, wherein the serial control data signal comprises an amount of the data to be written during a write operation.

124. The latency-independent interface of claim 121, wherein the serial control data signal comprises an amount of the data to be read during a read operation.

125. The latency-independent interface of claim 121, wherein the serial control data signal comprises a codeword size of a current sector.

126. The latency-independent interface of claim 121, wherein the serial control data signal comprises a information if a succeeding serial control data is a continuation of a current serial control data.

127. The latency-independent interface of claim 121, wherein the serial control data signal comprises:

during a write operation information as to a start of a sync mark and a start of write padding data, and

during a read operation information that a sync mark was detected.

128. The latency-independent interface of claim 121, wherein the first component further comprises a first ready transceiver that transmits or receives a bi-directional ready signal and wherein the second component further comprising a second ready transceiver that transmits or receives the bi-directional ready signal.

129. The latency-independent interface of claim 121, wherein said first hardware component comprises a disk controller and said second hardware component comprises a read channel.

130. The latency-independent interface of claim 121, wherein the first component further comprises a first sync mark transceiver that transmits or receives sync mark information and wherein the second component further comprises a second sync mark transceiver that transmits or receives the sync mark information.

131. The latency-independent interface of claim 130, wherein during a write operation a first assertion of the sync mark information indicates a start of sync mark

insertion and a second assertion of the sync mark information indicates a start of writing of padding data.

132. A data transmission system comprising:

a first component comprising:

a serial control transmitter circuit that transmits a serial control data signal; and

a first data transceiver circuit that transmits or receives data under the control of the serial control data signal,

a second component comprising:

a serial control receiver circuit that receives the serial control data signal; and

a second data transceiver circuit that transmits or receives the data under the control of the serial control data signal,

wherein during a write operation a first assertion by said first hardware component of the sync mark information indicates a start of sync mark insertion and a second assertion by said first hardware component of the sync mark information indicates a start of writing of padding data, and

during a read operation by said second hardware component information that a sync mark was detected.

133. The latency-independent interface of claim 132, wherein said first hardware component comprises a disk controller and said second hardware component comprises a read channel.

134. A data transmission system comprising:

a first component comprising:

a serial control transmitter circuit that transmits a serial control data signal;

a first data transceiver circuit that transmits or receives data under the control of the serial control data signal; and

a first ready transceiver that transmits or receives a ready signal,

a second component comprising:

a serial control receiver circuit that receives the serial control data signal; and

a second data transceiver circuit that transmits or receives the data under the control of the serial control data signal;

a second ready transceiver that transmits or receives the ready signal,

wherein during a write operation the ready signal indicates said second hardware component is ready to receive data from the first hard component; and

wherein during a read operation the ready signal indicates said first hardware component is ready to receive data from the second hard.

135. A data transmission system comprising:

first component means comprising:

serial control transmitter means for transmitting a serial control data signal; and

first data transceiver means for transmitting or receiving data under the control of the serial control data signal,

second component means comprising:

serial control receiver means for receiving the serial control data signal; and

a second data transceiver means for transmitting or receiving the data under the control of the serial control data signal,

wherein the serial control data signal comprises information as to whether the data is one of split and non-split.

136. The latency-independent interface of claim 135, wherein the serial control data signal comprises information that the data is one of first split, continue split and last split.

137. The latency-independent interface of claim 135, wherein the serial control data signal comprises an amount of the data to be written during a write operation.

138. The latency-independent interface of claim 135, wherein the serial control data signal comprises an amount of the data to be read during a read operation.

139. The latency-independent interface of claim 135, wherein the serial control data signal comprises a codeword size of a current sector.

140. The latency-independent interface of claim 135, wherein the serial control data signal comprises a information if a succeeding serial control data is a continuation of a current serial control data.

141. The latency-independent interface of claim 135, wherein the serial control data signal comprises

during a write operation information as to a start of a sync mark and a start of write padding data, and

during a read operation information that a sync mark was detected.

142. The latency-independent interface of claim 135, wherein said first component means further comprises first ready transceiver means for transmitting or receiving a

bi-directional ready signal, and wherein said second component means further comprises second ready transceiver means for transmitting or receiving the bi-directional ready signal.

143. The latency-independent interface of claim 135, wherein said first hardware component comprises disk controller means and said second hardware component comprises read channel means.

144. The latency-independent interface of claim 135, wherein said first component means further comprises first sync mark transceiver means for transmitting or receiving sync mark information, and wherein said second component means further comprises second sync mark transceiver means for transmitting or receiving the sync mark information.

145. The latency-independent interface of claim 144, wherein during a write operation a first assertion of the sync mark information indicates a start of sync mark insertion and a second assertion of the sync mark information indicates a start of writing of padding data.

146. A data transmission system comprising:

first component means comprising:

serial control transmitter means for transmitting a serial control data signal; and

first data transceiver means for transmitting or receiving data under the control of the serial control data signal,

second component means comprising:

serial control receiver means for receiving the serial control data signal; and

second data transceiver means for transmitting or receiving the data under the control of the serial control data signal,

wherein during a write operation a first assertion by said first hardware component means of the sync mark information indicates a start of sync mark insertion and a second assertion by said first hardware component means of the sync mark information indicates a start of writing of padding data, and

during a read operation by said second hardware component means information that a sync mark was detected.

147. The latency-independent interface of claim 146, wherein said first hardware component means comprises disk controller means and said second hardware component means comprises read channel means.

148. A data transmission system comprising:

first component means comprising:

serial control transmitter means for transmitting a serial control data signal;

first data transceiver means for transmitting or receiving data under the control of the serial control data signal; and

first ready transceiver for transmitting or receiving a ready signal,

second component means comprising:

serial control receiver means for receiving the serial control data signal; and

second data transceiver means for transmitting or receiving the data under the control of the serial control data signal;

second ready transceiver for transmitting or receiving the ready signal,

wherein during a write operation the ready signal indicates said second hardware component means is ready to receive data from the first hard component means; and

wherein during a read operation the ready signal indicates said first hardware component means is ready to receive data from the second hard.

149. The latency-independent interface of claim 148, wherein said first hardware component means comprises disk controller means and said second hardware component means comprises read channel means.

150. The latency-independent interface of claim 134, wherein said first hardware component means comprises a disk controller and said second hardware component means comprises a read channel.

151. A method of transmitting and receiving signals between first and second hardware components, comprising the steps of:

transmitting a serial control data signal from the first component;

receiving the serial control data signal by the second component;

transmitting or receiving data under the control of the serial control data signal by the first component; and

transmitting or receiving the data under the control of the serial control data signal by the second component,

wherein the serial control data signal comprises information as to whether the data is one of split and non-split.

152. The method of claim 151, wherein the serial control data signal comprises information that the data is one of first split, continue split and last split.

153. The method of claim 151, wherein the serial control data signal comprises an amount of the data to be written during a write operation.

154. The method of claim 151, wherein the serial control data signal comprises an amount of the data to be read during a read operation.

155. The method of claim 151, wherein the serial control data signal comprises a codeword size of a current sector.

156. The method of claim 151, wherein the serial control data signal comprises information if a succeeding serial control data is a continuation of a current serial control data.

157. The method of claim 151, wherein the serial control data signal comprises

during a write operation information as to a start of a sync mark and a start of write padding data, and

during a read operation information that a sync mark was detected.

158. The method of claim 151, further comprising the step of transmitting or receiving a bi-directional ready signal by the first component and transmitting or receiving the bi-directional ready signal by the second component.

159. The method of claim 151, wherein said first hardware component comprises a disk controller and said second hardware component comprises a read channel.

160. The method of claim 151, further comprising the step of transmitting or receiving sync mark information by the first component, and transmitting or receiving the sync mark information by the second component.

161. The method of claim 160, wherein during a write operation a first assertion of the sync mark information indicates a start of sync mark insertion and a second assertion of the sync mark information indicates a start of writing of padding data.

162. A method of transmitting and receiving signals between first and second hardware components, comprising the steps of:

transmitting a serial control data signal by the first component;

transmitting the serial control data signal by the second component;

transmitting or receiving data under the control of the serial control data signal by the first component;

transmitting or receiving the data under the control of the serial control data signal by the second component;

transmitting or receiving sync mark information by the first component; and

transmitting or receiving sync the mark information by the second component,

wherein during a write operation a first assertion by the first hardware component of the sync mark information indicates a start of sync mark insertion and a second assertion by the first hardware component of the sync mark information indicates a start of writing of padding data, and

during a read operation by the second hardware component information that a sync mark was detected.

163. The method of claim 162, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

164. A method of transmitting and receiving signals between first and second hardware components, comprising the steps of:

transmitting a serial control data signal by the first component;

transmitting the serial control data signal by the second component;

transmitting or receiving data under the control of the serial control data signal by the first component;

transmitting or receiving data under the control of the serial control data signal by the second component;

transmitting or receiving a ready signal by the first component; and

transmitting or receiving the ready signal by the second component;

during a write operation the ready signal indicates said second hardware component is ready to receive data from the first hard component; and

during a read operation the ready signal indicates said first hardware component is ready to receive data from the second hard component.

165. The method of claim 164, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

166. A latency-independent interface between a first hardware component having a controller and a second hardware component, comprising:

a clock circuit that receives a clock signal;

a data circuit that transmits or receives data;

a sync mark receive circuit to receive a sync mark signal; and

a data gate circuit that transmits a data gate signal,

wherein during a read operation after the data gate signal is transmitted, the controller counts the clock signal when the sync mark is received and when the count is equal the amount of data to be read, a time period is dropped from the clock signal.

167. A latency-independent interface according to Claim 166, wherein the time period is one cycle.

168. A latency-independent interface according to Claim 166, wherein after the count is equal to the amount of data to be read, the controller transmits the length of the data to be read.

169. A latency-independent interface according to Claim 166, wherein the data comprises a header indicating a number of sync fields for each data gate signal and a tail indicating bytes to be written for each data gate signal.

170. A latency-independent interface between a first hardware component having a controller and a second hardware component, comprising:

a clock circuit that transmits a clock signal;

a data circuit that transmits or receives data;

a sync mark transmit circuit to transmit a sync mark signal; and

a data gate circuit that receives a data gate signal,

wherein during a read operation after the data gate signal is received, the controller counts the clock signal when the sync mark is received and when the count is equal the amount of data to be read, a time period is dropped from the clock signal.

171. A latency-independent interface according to Claim 170, wherein the time period is one cycle.

172. A latency-independent interface according to Claim 170, wherein after the count is equal to the amount of data to be read, the controller transmits the length of the data to be read.

173. A latency-independent interface according to Claim 170, wherein the data comprises a header indicating a number of sync fields for each data gate signal and a tail indicating bytes to be written for each data gate signal.

174. A latency-independent interface between a first hardware component having a controller and a second hardware component, comprising:

a data circuit that transmits or receives data; and

a parser circuit,

wherein said parser circuit parses the data comprising a header indicating a number of sync fields for each data gate signal and a tail indicating bytes to be written for each data gate signal.

175. The latency-independent interface of claim 166, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

176. The latency-independent interface of claim 170, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

177. The latency-independent interface of claim 174, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

178. A data transmission system comprising:

a first hardware component comprising:

a controller;

a first clock circuit that receives a clock signal;

a first data circuit that transmits or receives data;

a first sync mark receive circuit to receive a sync mark signal; and

a first data gate circuit that transmits a data gate signal; and

a second hardware component comprising:

a second clock circuit that transmits the clock signal;

a second data circuit that transmits or receives the data;

a second sync mark transmit circuit to transmit the sync mark signal; and

a second data gate circuit that receives the data gate signal; and

wherein during a read operation after the data gate signal is transmitted, said controller counts the clock signal when the sync mark is received and when the count is equal the amount of data to be read, a time period is dropped from the clock signal.

179. A latency-independent interface according to Claim 178, wherein the time period is one cycle.

180. A latency-independent interface according to Claim 178, wherein after the count is equal to the amount of data to be read, said controller transmits the length of the data to be read.

181. A latency-independent interface according to Claim 178, wherein the data comprises a header indicating a number of sync fields for each data gate signal and a tail indicating bytes to be written for each data gate signal.

182. The latency-independent interface of claim 181, wherein said first hardware component comprises a disk controller and said second hardware component comprises a read channel.

183. A data transmission system comprising:

a first hardware component comprising:

a controller; and

a first data circuit that transmits or receives data; and

a second hardware component, comprising:

a second data circuit that transmits or receives the data; and

a parser circuit,

wherein said parser circuit parses the data comprising a header indicating a number of sync fields for each data gate signal and a tail indicating bytes to be written for each data gate signal.

184. The latency-independent interface of claim 183, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

185. A latency-independent interface between a first hardware component having a controller and a second hardware component, comprising:

clock means for receiving a clock signal;

transceiver means for transmitting or receiving data;

sync mark receiving means to receive a sync mark signal; and

data gate means for transmitting a data gate signal,

wherein during a read operation after the data gate signal is transmitted, the controller counts the clock signal when the sync mark is received and when the count is equal the amount of data to be read, a time period is dropped from the clock signal.

186. A latency-independent interface according to Claim 185, wherein the time period is one cycle.

187. A latency-independent interface according to Claim 185, wherein after the count is equal to the amount of data to be read, the controller transmits the length of the data to be read.

188. A latency-independent interface according to Claim 185, wherein the data comprises a header indicating a number of sync fields for each data gate signal and a tail indicating bytes to be written for each data gate signal.

189. A latency-independent interface between a first hardware component having a controller and a second hardware component, comprising:

clock means for transmitting a clock signal;

transceiver means for transmitting or receiving data;

sync mark transmitting means for transmitting a sync mark signal; and

data gate means for receiving a data gate signal,

wherein during a read operation after the data gate signal is received, the controller counts the clock signal when the sync mark is received and when the count is equal the amount of data to be read, a time period is dropped from the clock signal.

190. A latency-independent interface according to Claim 189, wherein the time period is one cycle.

191. A latency-independent interface according to Claim 189, wherein after the count is equal to the amount of data to be read, the controller transmits the length of the data to be read.

192. A latency-independent interface according to Claim 189, wherein the data comprises a header indicating a number of sync fields for each data gate signal and a tail indicating bytes to be written for each data gate signal.

193. A latency-independent interface between a first hardware component having a controller and a second hardware component, comprising:

transceiver means for transmitting or receiving data; and

parser means for parsing the data,

wherein said parser means parses the data comprising a header indicating a number of sync fields for each data gate signal and a tail indicating bytes to be written for each data gate signal.

194. The latency-independent interface of claim 185, wherein the first hardware component comprises disk controller means and the second hardware component comprises a read channel means.

195. The latency-independent interface of claim 189, wherein the first hardware component comprises disk controller means and the second hardware component comprises read channel means.

196. The latency-independent interface of claim 193, wherein the first hardware component comprises disk controller and the second hardware component comprises read channel means.

197. A data transmission system comprising:

first hardware means comprising:

controller means for controlling said first hardware component;

first clock means for receiving a clock signal;

first transceiver means for transmitting or receiving data;

first sync mark receiving means to receive a sync mark signal; and

first data gate means for transmitting a data gate signal; and

second hardware means comprising:

second clock means for transmitting the clock signal;

second transceiver means for transmitting or receiving the data;

second sync mark transmitting means for transmitting the sync mark signal; and

second data gate means for receiving the data gate signal; and

wherein during a read operation after the data gate signal is transmitted, said controller means counts the clock signal when the sync mark is received and when the count is equal the amount of data to be read, a time period is dropped from the clock signal.

198. A latency-independent interface according to Claim 197, wherein the time period is one cycle.

199. A latency-independent interface according to Claim 197, wherein after the count is equal to the amount of data to be read, said controller means transmits the length of the data to be read.

200. A latency-independent interface according to Claim 197, wherein the data comprises a header indicating a number of sync fields for each data gate signal and a tail indicating bytes to be written for each data gate signal.

200. The latency-independent interface of claim 197, wherein said first hardware means comprises disk controller means and said second hardware means comprises read channel means.

201. A data transmission system comprising:

first hardware means comprising:

controller means for controlling said first hardware means, and

first transceiver means for transmitting or receiving data; and

second hardware means, comprising:

second transceiver means for transmitting or receiving the data; and

parser means,

wherein said parser means parses the data comprising a header indicating a number of sync fields for each data gate signal and a tail indicating bytes to be written for each data gate signal.

202. The latency-independent interface of claim 201, wherein the first hardware means comprises disk controller means and the second hardware means comprises read channel means.

203. A method of transmitting and receiving signals between a first hardware component having a controller and a second hardware component, comprising the steps of:

receiving a clock signal;

transmitting or receiving data;

receiving a sync mark signal; and

transmitting a data gate signal,

wherein during a read operation after the data gate signal is transmitted, the clock signal is counted when the sync mark is received and when the count is equal the amount of data to be read, a time period is dropped from the clock signal.

204. A method according to Claim 203, wherein the time period is one cycle.

205. A method according to Claim 203, wherein after the count is equal to the amount of data to be read, the controller transmits the length of the data to be read.

206. A method according to Claim 203, wherein the data comprises a header indicating a number of sync fields for each data gate signal and a tail indicating bytes to be written for each data gate signal.

207. A method of transmitting and receiving signals between a first hardware component having a controller and a second hardware component, comprising the steps of:

clock means for transmitting a clock signal;

transceiver means for transmitting or receiving data;

transmitting a sync mark signal; and

receiving a data gate signal,

wherein during a read operation after the data gate signal is received, the clock signal is counted when the sync mark is received and when the count is equal the amount of data to be read, a time period is dropped from the clock signal.

208. A method according to Claim 207, wherein the time period is one cycle.

209. A method according to Claim 207, wherein after the count is equal to the amount of data to be read, transmitting the length of the data to be read.

210. A method according to Claim 207, wherein the data comprises a header indicating a number of sync fields for each data gate signal and a tail indicating bytes to be written for each data gate signal.

211. A method of transmitting and receiving signals between a first hardware component having a controller and a second hardware component, comprising the steps of:

transmitting or receiving data; and

parsing the data, the data comprising a header indicating a number of sync fields for each data gate signal and a tail indicating bytes to be written for each data gate signal.

212. A computer program for transmitting and receiving signals between first and second hardware components, comprising the steps of:

transmitting a serial control data signal from the first component;

receiving the serial control data signal by the second component;

transmitting or receiving data under the control of the serial control data signal by the first component; and

transmitting or receiving the data under the control of the serial control data signal by the second component,

wherein the serial control data signal comprises information as to whether the data is one of split and non-split.

213. The computer program of claim 212, wherein the serial control data signal comprises information that the data is one of first split, continue split and last split.

214. The computer program of claim 212, wherein the serial control data signal comprises an amount of the data to be written during a write operation.

215. The computer program of claim 212, wherein the serial control data signal comprises an amount of the data to be read during a read operation.

216. The computer program of claim 212, wherein the serial control data signal comprises a codeword size of a current sector.

217. The computer program of claim 212, wherein the serial control data signal comprises information if a succeeding serial control data is a continuation of a current serial control data.

218. The computer program of claim 212, wherein the serial control data signal comprises

during a write operation information as to a start of a sync mark and a start of write padding data, and

during a read operation information that a sync mark was detected.

219. The computer program of claim 212, further comprising the step of transmitting or receiving a bi-directional ready signal by the first component and transmitting or receiving the bi-directional ready signal by the second component.

220. The computer program of claim 212, wherein said first hardware component comprises a disk controller and said second hardware component comprises a read channel.

221. The computer program of claim 212, further comprising the step of transmitting or receiving sync mark information by the first component, and transmitting or receiving the sync mark information by the second component.

222. The computer program of claim 221, wherein during a write operation a first assertion of the sync mark information indicates a start of sync mark insertion and a second assertion of the sync mark information indicates a start of writing of padding data.

223. A computer program for transmitting and receiving signals between first and second hardware components, comprising the steps of:

transmitting a serial control data signal by the first component;

transmitting the serial control data signal by the second component;

transmitting or receiving data under the control of the serial control data signal by the first component;

transmitting or receiving the data under the control of the serial control data signal by the second component;

transmitting or receiving sync mark information by the first component; and

transmitting or receiving sync the mark information by the second component,

wherein during a write operation a first assertion by the first hardware component of the sync mark information indicates a start of sync mark insertion and a second assertion by the first hardware component of the sync mark information indicates a start of writing of padding data, and

during a read operation by the second hardware component information that a sync mark was detected.

224. The computer program of claim 223, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

225. A computer program for transmitting and receiving signals between first and second hardware components, comprising the steps of:

transmitting a serial control data signal by the first component;

transmitting the serial control data signal by the second component;

transmitting or receiving data under the control of the serial control data signal by the first component;

transmitting or receiving data under the control of the serial control data signal by the second component;

transmitting or receiving a ready signal by the first component; and

transmitting or receiving the ready signal by the second component;

during a write operation the ready signal indicates said second hardware component is ready to receive data from the first hard component; and

during a read operation the ready signal indicates said first hardware component is ready to receive data from the second hard component.

225. The computer program of claim 224, wherein the first hardware component comprises a disk controller and the second hardware component comprises a read channel.

226. A computer program for transmitting and receiving signals between a first hardware component having a controller and a second hardware component, comprising the steps of:

receiving a clock signal;

transmitting or receiving data;

receiving a sync mark signal; and

transmitting a data gate signal,

wherein during a read operation after the data gate signal is transmitted, the clock signal is counted when the sync mark is received and when the count is equal the amount of data to be read, a time period is dropped from the clock signal.

227. A method according to Claim 226, wherein the time period is one cycle.

228. A method according to Claim 226, wherein after the count is equal to the amount of data to be read, the controller transmits the length of the data to be read.

229. A method according to Claim 226, wherein the data comprises a header indicating a number of sync fields for each data gate signal and a tail indicating bytes to be written for each data gate signal.

230. A computer program for transmitting and receiving signals between a first hardware component having a controller and a second hardware component, comprising the steps of:

clock means for transmitting a clock signal;

transceiver means for transmitting or receiving data;

transmitting a sync mark signal; and

receiving a data gate signal,

wherein during a read operation after the data gate signal is received, the clock signal is counted when the sync mark is received and when the count is equal the amount of data to be read, a time period is dropped from the clock signal.

231. A method according to Claim 230, wherein the time period is one cycle.

232. A method according to Claim 230, wherein after the count is equal to the amount of data to be read, transmitting the length of the data to be read.

233. A method according to Claim 230, wherein the data comprises a header indicating a number of sync fields for each data gate signal and a tail indicating bytes to be written for each data gate signal.

231. A computer program for transmitting and receiving signals between a first hardware component having a controller and a second hardware component, comprising the steps of:

transmitting or receiving data; and

parsing the data, the data comprising a header indicating a number of sync fields for each data gate signal and a tail indicating bytes to be written for each data gate signal.

232. A latency-independent interface between first and second hardware components, comprising:

a read/write control circuit to receive a read/write control signal;

a data valid control circuit to receive a data valid control signal;

a read/write gate circuit to receive a read/write gate control signal;

a data transceiver to transmit and receive user data; and

an external write gate circuit to provide an external write gate signal,

wherein during a write operation the read/write signal indicates a write operation,

wherein at a first time the data valid control signal is asserted for a first period of time less than a codeword of the user data, and

wherein at a second time the read/write gate control signal is asserted, the external write gate circuit is asserted for.second period of time greater than plural codeword of the user data.

233. A read/write channel for a storage media, comprising:

a read/write control circuit to receive a read/write control signal;

a data valid control circuit to receive a data valid control signal;

a read/write gate circuit to receive a read/write gate control signal;

a data transceiver to transmit and receive user data; and

an external write gate circuit to provide an external write gate signal,

wherein during a write operation the read/write signal indicates a write operation,

wherein at a first time the data valid control signal is asserted for a first period of time less than a codeword of the user data, and

wherein at a second time the read/write gate control signal is asserted, the external write gate circuit is asserted for.second period of time greater than plural codeword of the user data.

234. A latency-independent interface between first and second hardware components, comprising:

a read/write control circuit to receive a read/write control signal;

a data valid control circuit to provide a data valid control signal;

a read/write gate circuit to receive a read/write gate control signal;

a data transceiver to transmit and receive user data; and

an internal gate circuit to provide an internal read gate signal,

wherein during a read operation the read/write signal indicates a read operation,

wherein at a first time the read/write gate is asserted for a first time period,

wherein at the first time the internal read gate is asserted for a second time period greater than the first time period,

wherein at a second time the data valid control signal is asserted for a third period of time greater than a codeword of the user data.

235. A read/write channel for a storage media, comprising:

a read/write control circuit to receive a read/write control signal;

a data valid control circuit to provide a data valid control signal;

a read/write gate circuit to receive a read/write gate control signal;

a data transceiver to transmit and receive user data; and

an internal gate circuit to provide an internal read gate signal,

wherein during a read operation the read/write signal indicates a read operation,

wherein at a first time the read/write gate is asserted for a first time period,

wherein at the first time the internal read gate is asserted for a second time period greater than the first time period,

wherein at a second time the data valid control signal is asserted for a third period of time greater than a codeword of the user data.

236. A latency-independent interface between first and second hardware components, comprising:

read/write control means for receiving a read/write control signal;

data valid control means for receiving a data valid control signal;

read/write gate means for receiving a read/write gate control signal;

data transceiver means for transmitting and receiving user data; and

external write gate means for providing an external write gate signal,

wherein during a write operation the read/write signal indicates a write operation,

wherein at a first time the data valid control signal is asserted for a first period of time less than a codeword of the user data, and

wherein at a second time the read/write gate control signal is asserted, the external write gate means is asserted for.second period of time greater than plural codeword of the user data.

237. A read/write channel for a storage media, comprising:

read/write control means for receiving a read/write control signal;

data valid control means for receiving a data valid control signal;

read/write gate means for receiving a read/write gate control signal;

data transceiver means for transmitting and receiving user data; and

an external write gate means for providing an external write gate signal,

wherein during a write operation the read/write signal indicates a write operation,

wherein at a first time the data valid control signal is asserted for a first period of time less than a codeword of the user data, and

wherein at a second time the read/write gate control signal is asserted, the external write gate means is asserted for.second period of time greater than plural codewords of the user data.

238. A latency-independent interface between first and second hardware components, comprising:

read/write control means for receiving a read/write control signal;

data valid control means for providing a data valid control signal;

read/write gate means for receiving a read/write gate control signal;

data transceiver means for transmitting and receiving user data; and

internal gate means for providing an internal read gate signal,

wherein during a read operation the read/write signal indicates a read operation,

wherein at a first time the read/write gate is asserted for a first time period,

wherein at the first time the internal read gate is asserted for a second time period greater than the first time period,

wherein at a second time the data valid control signal is asserted for a third period of time greater than a codeword of the user data.

239. A read/write channel for a storage media, comprising:

read/write control means for receiving a read/write control signal;

data valid control means for providing a data valid control signal;

read/write gate means for receiving a read/write gate control signal;

data transceiver means for transmitting and receiving user data; and

internal gate means for providing an internal read gate signal,

wherein during a read operation the read/write signal indicates a read operation,

wherein at a first time the read/write gate is asserted for a first time period,

wherein at the first time the internal read gate is asserted for a second time period greater than the first time period,

wherein at a second time the data valid control signal is asserted for a third period of time greater than a codeword of the user data.

240. A method for interfacing between first and second hardware components, comprising:

receiving a read/write control signal;

receiving a data valid control signal;

receiving a read/write gate control signal;

transmitting and receiving user data; and

providing an external write gate signal,

wherein during a write operation the read/write signal indicates a write operation,

wherein at a first time the data valid control signal is asserted for a first period of time less than a codeword of the user data, and

wherein at a second time the read/write gate control signal is asserted, the external write gate means is asserted for.second period of time greater than plural codeword of the user data.

241. A method of reading from and writing from a storage media, comprising:

receiving a read/write control signal;

receiving a data valid control signal;

receiving a read/write gate control signal;

transmitting and receiving user data; and

providing an external write gate signal,

wherein during a write operation the read/write signal indicates a write operation,

wherein at a first time the data valid control signal is asserted for a first period of time less than a codeword of the user data, and

wherein at a second time the read/write gate control signal is asserted, the external write gate means is asserted for.second period of time greater than plural codewords of the user data.

242. A method for interfacing between first and second hardware, comprising:

receiving a read/write control signal;

providing a data valid control signal;

receiving a read/write gate control signal;

transmitting and receiving user data; and

providing an internal read gate signal,

wherein during a read operation the read/write signal indicates a read operation,

wherein at a first time the read/write gate is asserted for a first time period,

wherein at the first time the internal read gate is asserted for a second time period greater than the first time period,

wherein at a second time the data valid control signal is asserted for a third period of time greater than a codeword of the user data.

243. A method of reading from and writing from a storage media, comprising:

receiving a read/write control signal;

providing a data valid control signal;

receiving a read/write gate control signal;

transmitting and receiving user data; and

providing an internal read gate signal,

wherein during a read operation the read/write signal indicates a read operation,

wherein at a first time the read/write gate is asserted for a first time period,

wherein at the first time the internal read gate is asserted for a second time period greater than the first time period,

wherein at a second time the data valid control signal is asserted for a third period of time greater than a codeword of the user data.

244. A computer program for interfacing between first and second hardware components, comprising:

receiving a read/write control signal;

receiving a data valid control signal;

receiving a read/write gate control signal;

transmitting and receiving user data; and

providing an external write gate signal,

wherein during a write operation the read/write signal indicates a write operation,

wherein at a first time the data valid control signal is asserted for a first period of time less than a codeword of the user data, and

wherein at a second time the read/write gate control signal is asserted, the external write gate means is asserted for.second period of time greater than plural codeword of the user data.

245. A computer program reading from and writing from a storage media, comprising:

receiving a read/write control signal;

receiving a data valid control signal;

receiving a read/write gate control signal;

transmitting and receiving user data; and

providing an external write gate signal,

wherein during a write operation the read/write signal indicates a write operation,

wherein at a first time the data valid control signal is asserted for a first period of time less than a codeword of the user data, and

wherein at a second time the read/write gate control signal is asserted, the external write gate means is asserted for.second period of time greater than plural codewords of the user data.

246. A computer program for interfacing between first and second hardware, comprising:

receiving a read/write control signal;

providing a data valid control signal;

receiving a read/write gate control signal;

transmitting and receiving user data; and

providing an internal read gate signal,

wherein during a read operation the read/write signal indicates a read operation,

wherein at a first time the read/write gate is asserted for a first time period,

wherein at the first time the internal read gate is asserted for a second time period greater than the first time period,

wherein at a second time the data valid control signal is asserted for a third period of time greater than a codeword of the user data.

247. A computer program for reading from and writing from a storage media, comprising:

receiving a read/write control signal;

providing a data valid control signal;

receiving a read/write gate control signal;

transmitting and receiving user data; and

providing an internal read gate signal,

wherein during a read operation the read/write signal indicates a read operation,

wherein at a first time the read/write gate is asserted for a first time period,

wherein at the first time the internal read gate is asserted for a second time period greater than the first time period,

wherein at a second time the data valid control signal is asserted for a third period of time greater than a codeword of the user data.

UNITED STATES PATENT AND TRADEMARK OFFICE
DOCUMENT CLASSIFICATION BARCODE SHEET

Abstract

6

Level - 2
Version 1.1

ABSTRACT

A data transmission system is provided comprising a disk controller and a read channel. The disk controller comprises a controller for controlling said first hardware component, a first clock for receiving a clock signal, a first transceiver for transmitting or receiving data, a first sync mark receiver to receive a sync mark signal, and a first data gate circuit for transmitting a data gate signal. The read channel comprises a second clock for transmitting the clock signal, a second transceiver for transmitting or receiving the data, a second sync mark transmitter transmitting the sync mark signal, and a second data gate circuit for receiving the data gate signal. During a read operation after the data gate signal is transmitted, the controller counts the clock signal when the sync mark is received and when the count is equal the amount of data to be read, a time period is dropped from the clock signal.

Drawings

7

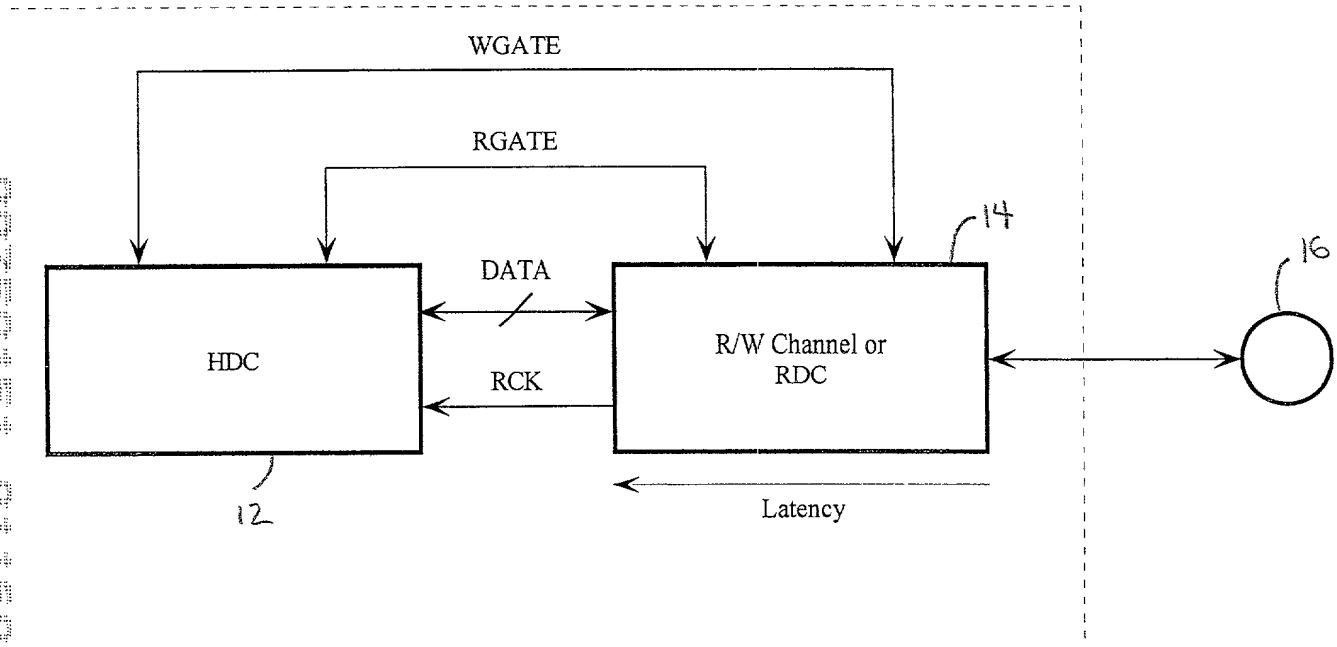


Fig. 1

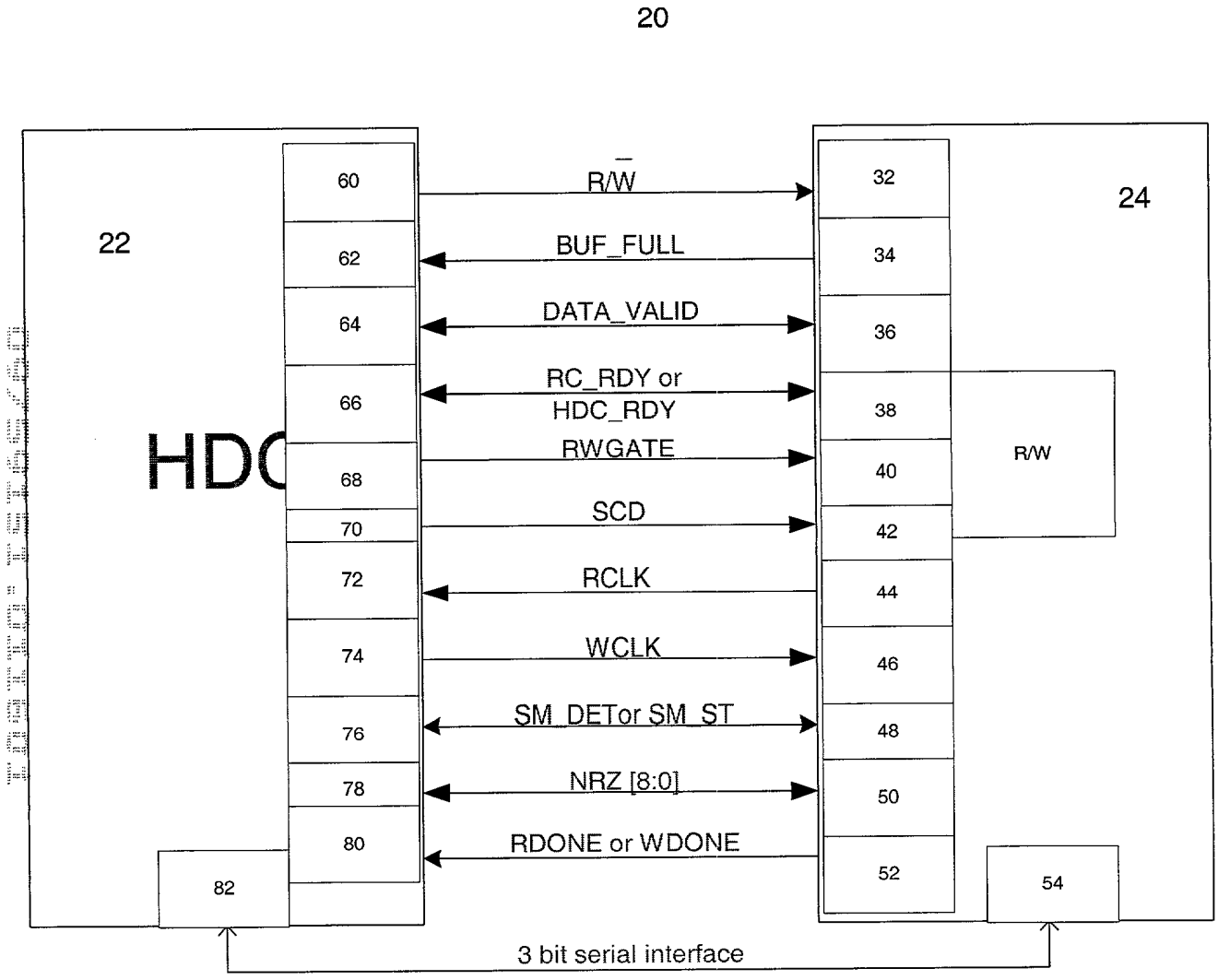

20
22
HDC
24
60
62
64
66
68
70
72
74
76
78
80
82
32
34
36
38
40
42
44
46
48
50
52
54
R/W
R/W
BUF_FULL
DATA_VALID
RC_RDY or
HDC_RDY
RWGATE
SCD
RCLK
WCLK
SM_DETor SM_ST
NRZ [8:0]
RDONE or WDONE
3 bit serial interface

Fig. 2

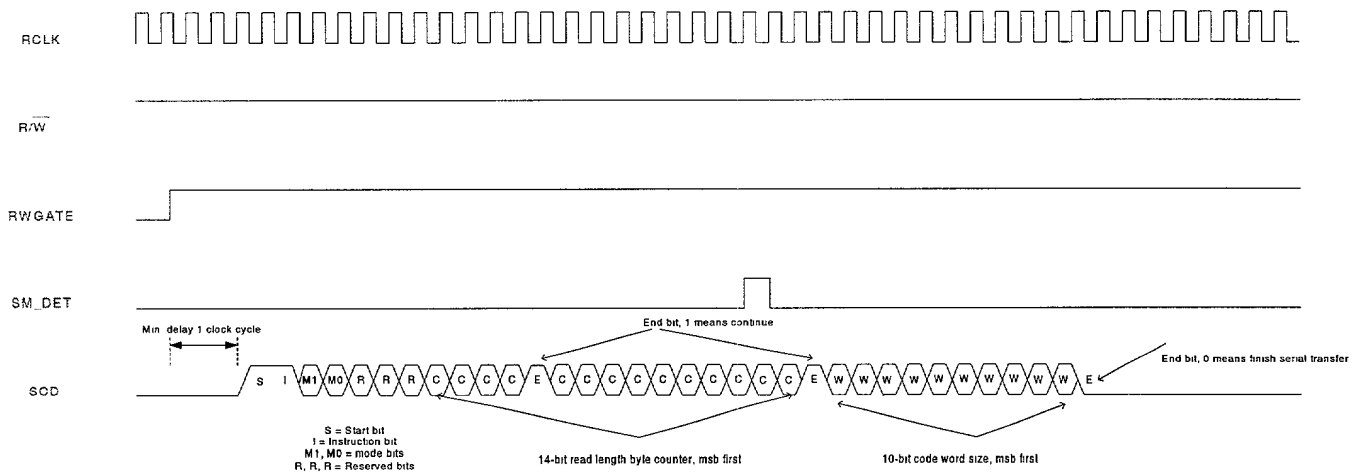

RCLK
R/W
RWGATE
SM_DET
SCD
Min delay 1 clock cycle
End bit, 1 means continue
End bit, 0 means finish serial transfer
S
I
M1
M0
R
R
R
C
E
W
E
S = Start bit
I = Instruction bit
M1, M0 = mode bits
R, R, R = Reserved bits
14-bit read length byte counter, msb first
10-bit code word size, msb first

Fig. 3

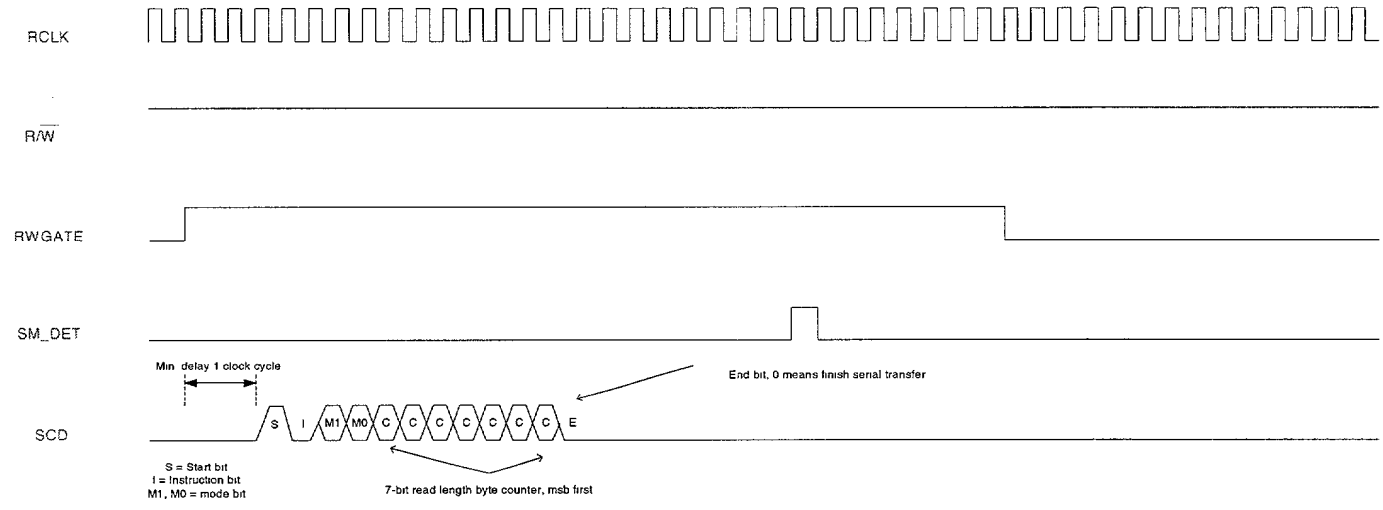

RCLK
R/W
RWGATE
SM_DET
SCD
Min delay 1 clock cycle
End bit, 0 means finish serial transfer
S
I
M1
M0
C
E
S = Start bit
I = Instruction bit
M1, M0 = mode bit
7-bit read length byte counter, msb first

Fig. 4

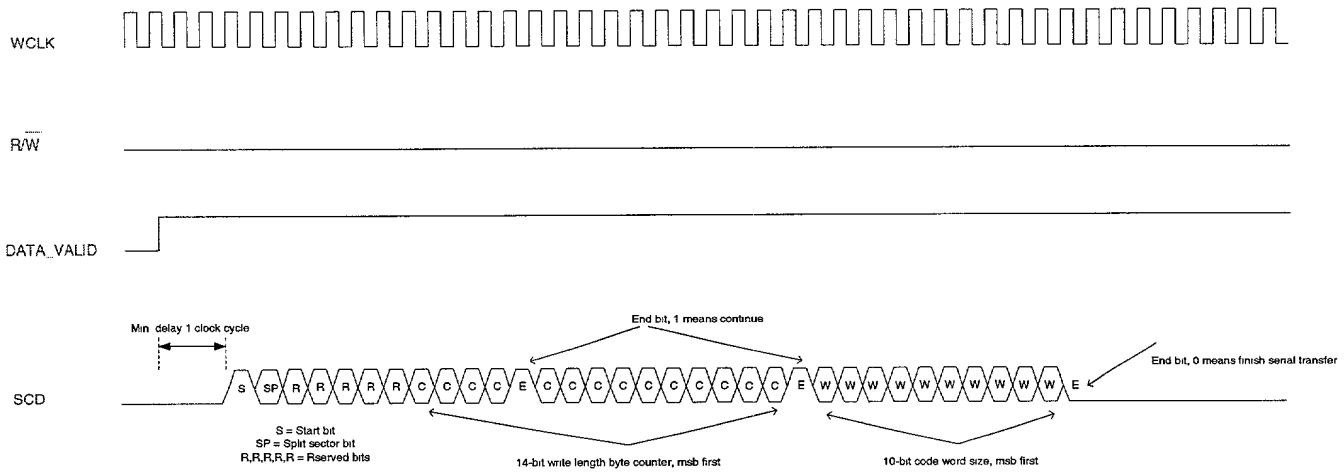

WCLK
R/W
DATA_VALID
SCD
Min delay 1 clock cycle
End bit, 1 means continue
End bit, 0 means finish serial transfer
S = Start bit
SP = Split sector bit
R,R,R,R,R = Rserved bits
14-bit write length byte counter, msb first
10-bit code word size, msb first

Fig. 5

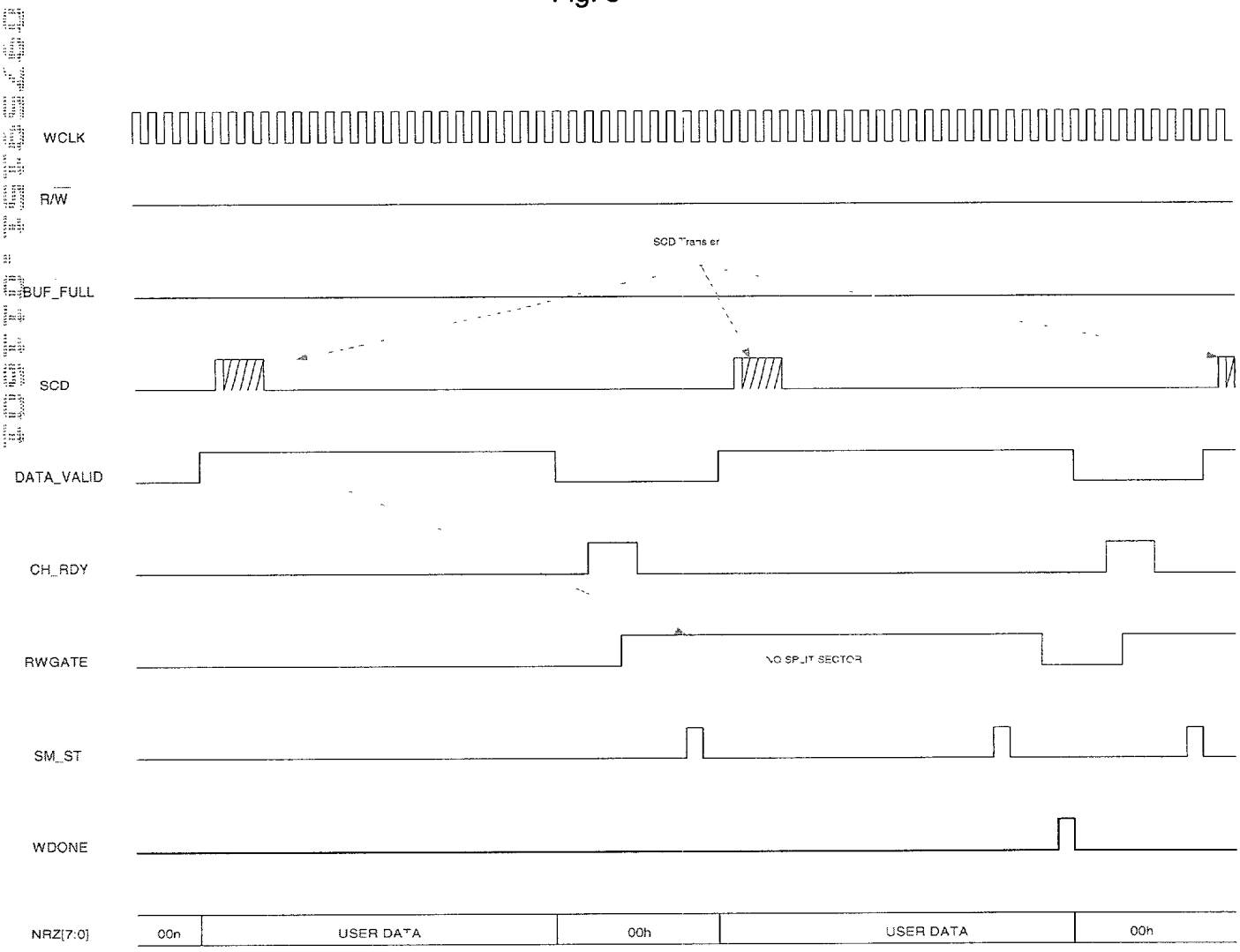

WCLK
R/W
BUF_FULL
SCD
DATA_VALID
CH_RDY
RWGATE
SM_ST
WDONE
NRZ[7:0]
00h
USER DATA
00h
USER DATA
00h

Fig. 6

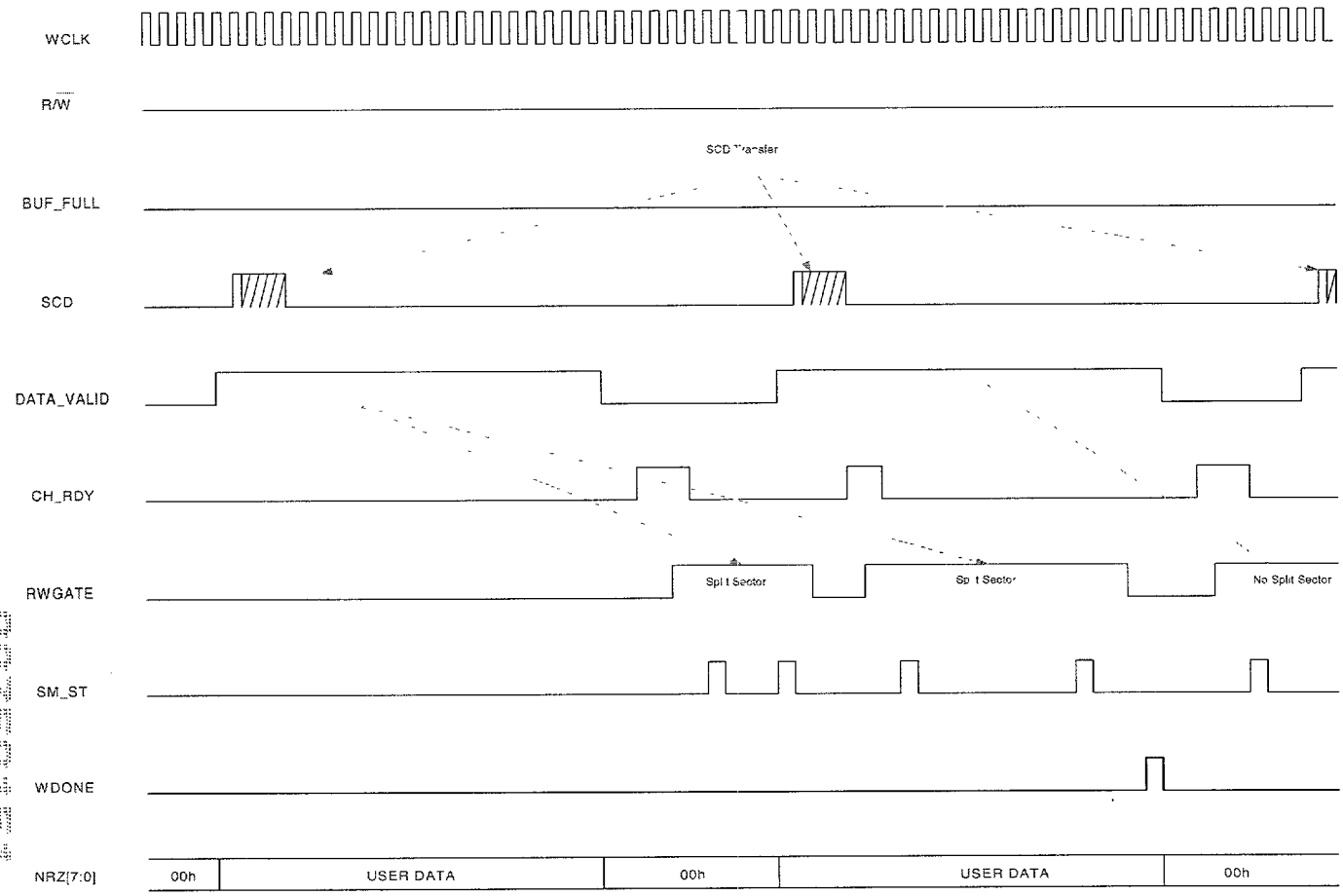

WCLK
R/W
BUF_FULL
SCD
DATA_VALID
CH_RDY
RWGATE
SM_ST
WDONE
NRZ[7:0]
Split Sector
Split Sector
No Split Sector
00h
USER DATA
00h
USER DATA
00h

Fig. 7

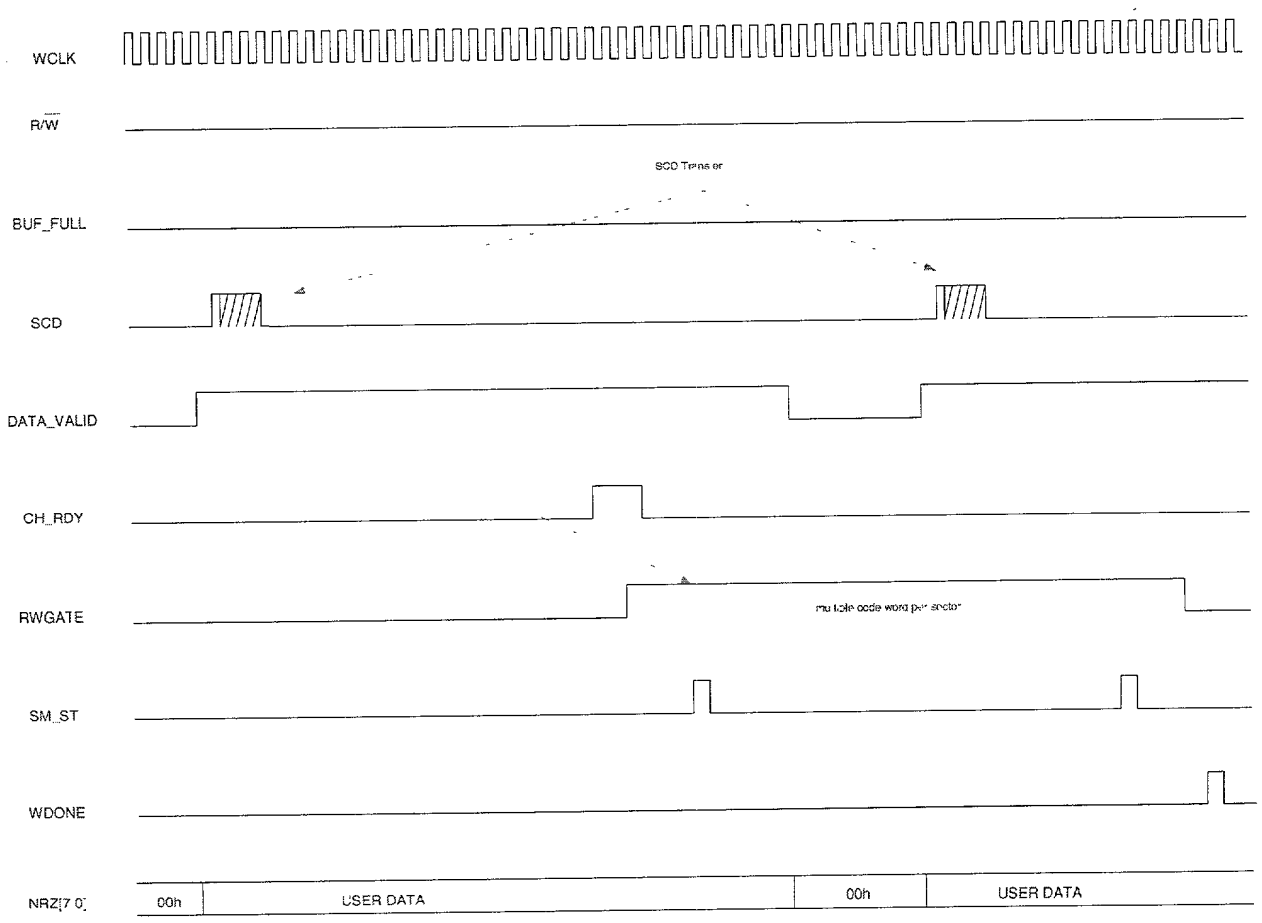

WCLK
R/W
SCD Transfer
BUF_FULL
SCD
DATA_VALID
CH_RDY
RWGATE
multiple code word per sector
SM_ST
WDONE
NRZ[7:0]
00h
USER DATA
00h
USER DATA

Fig. 8

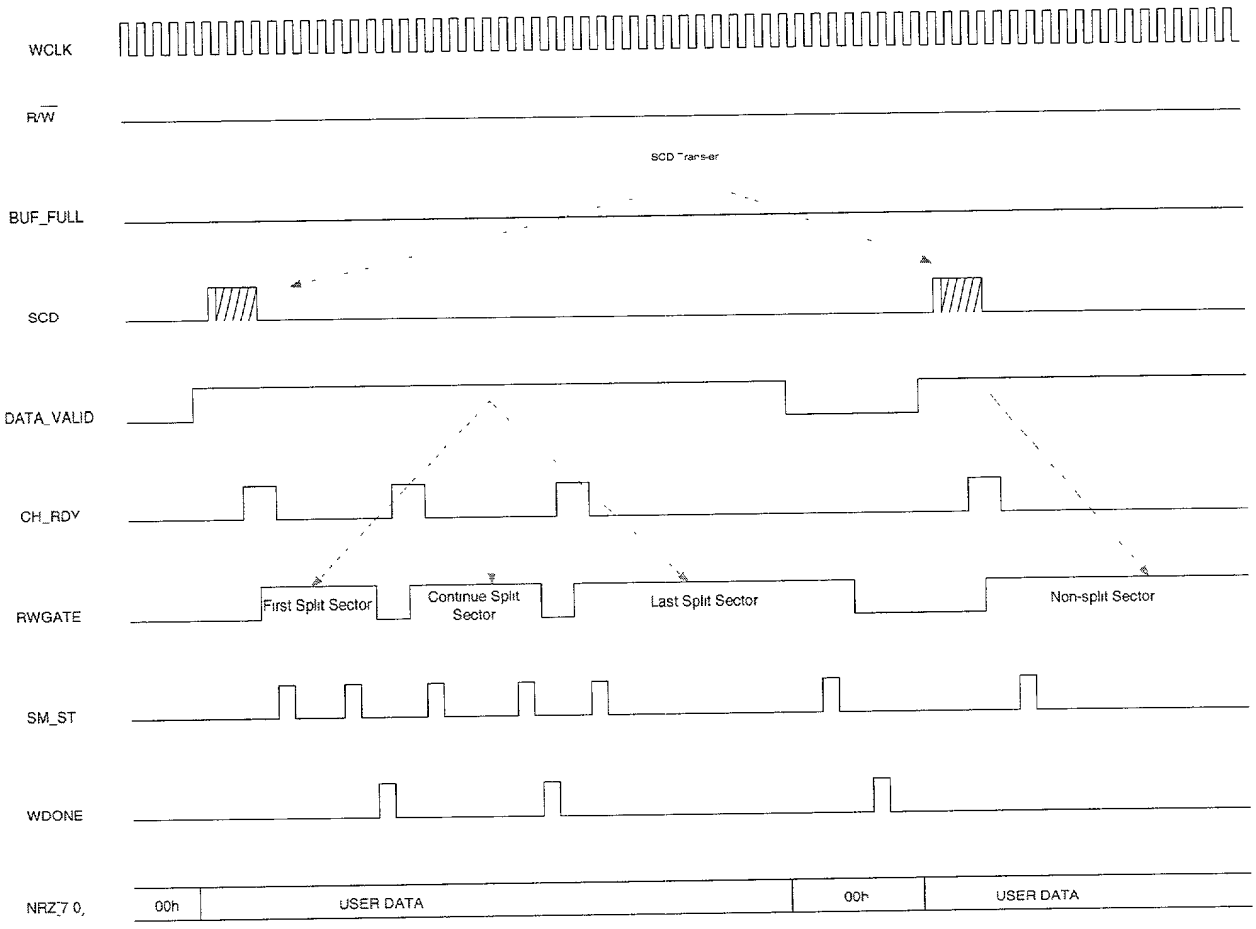

WCLK
R/W
SCD Transfer
BUF_FULL
SCD
DATA_VALID
CH_RDY
RWGATE
First Split Sector
Continue Split Sector
Last Split Sector
Non-split Sector
SM_ST
WDONE
NRZ_7 0_
00h
USER DATA
00h
USER DATA

Fig. 9

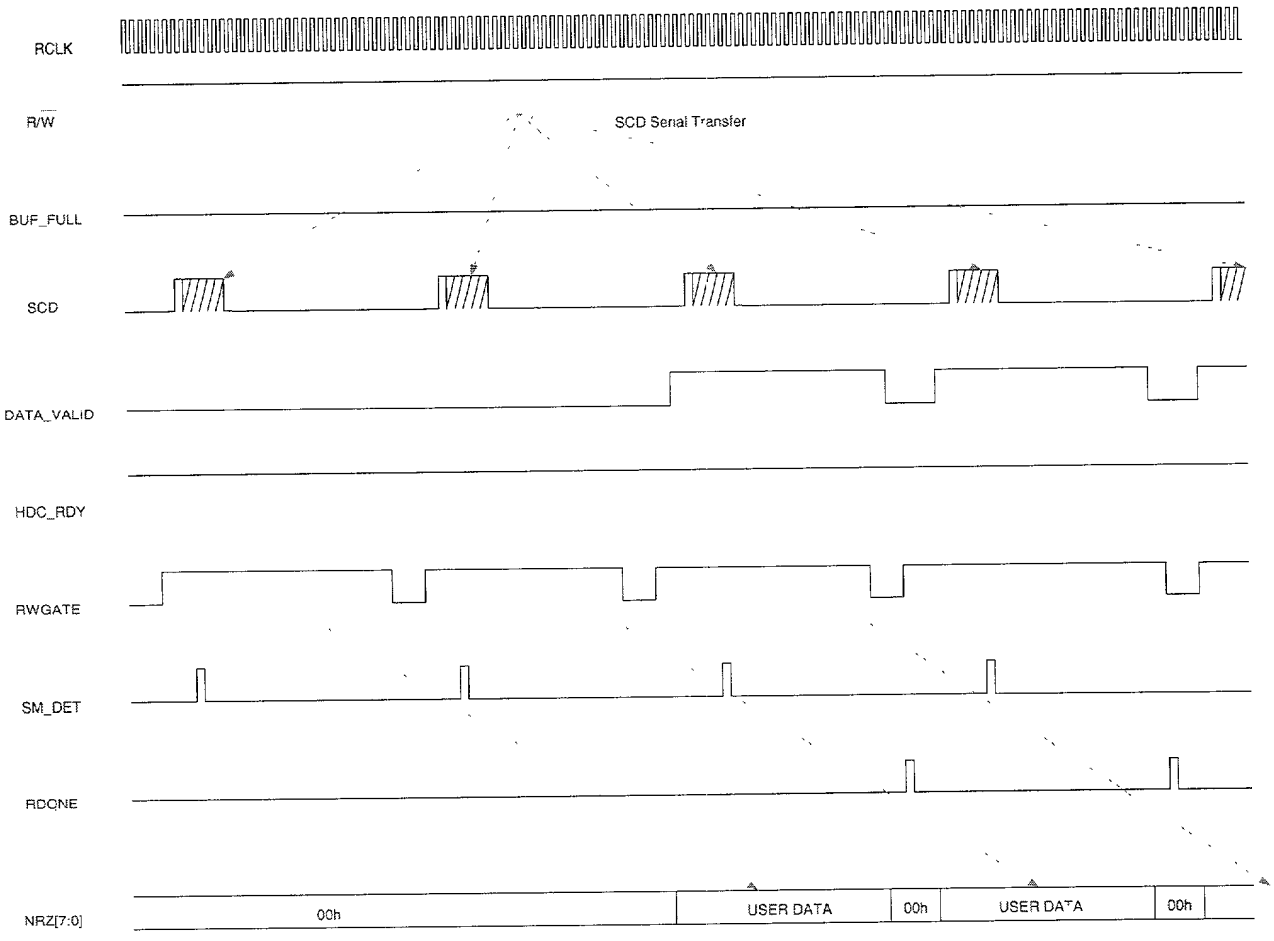

RCLK
R/W
SCD Serial Transfer
BUF_FULL
SCD
DATA_VALID
HDC_RDY
RWGATE
SM_DET
RDONE
NRZ[7:0]
00h
USER DATA
00h
USER DATA
00h

Fig. 10

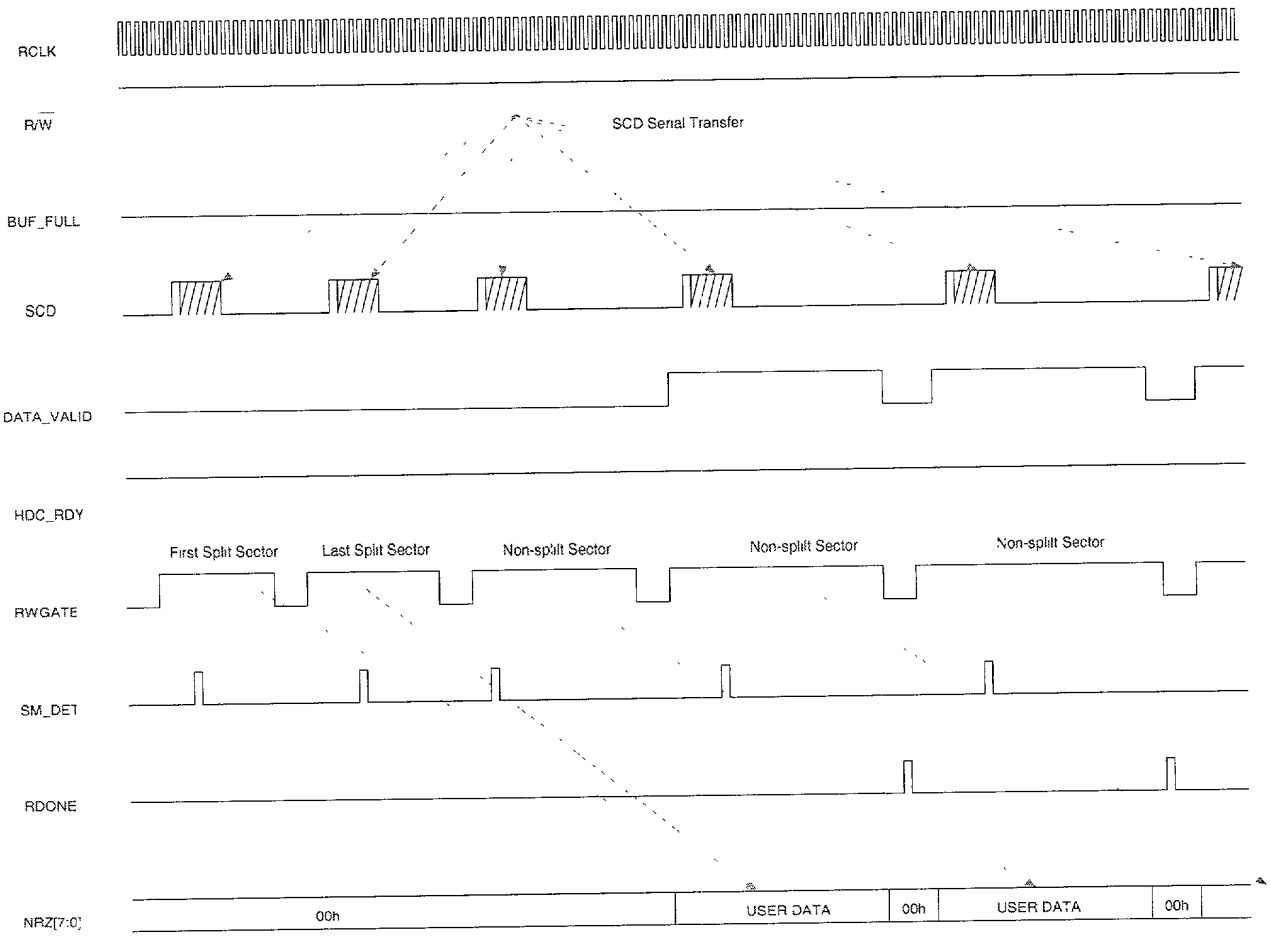

RCLK
R/W
SCD Serial Transfer
BUF_FULL
SCD
DATA_VALID
HDC_RDY
First Split Sector
Last Split Sector
Non-split Sector
Non-split Sector
Non-split Sector
RWGATE
SM_DET
RDONE
NRZ[7:0]
00h
USER DATA
00h
USER DATA
00h

Fig. 11

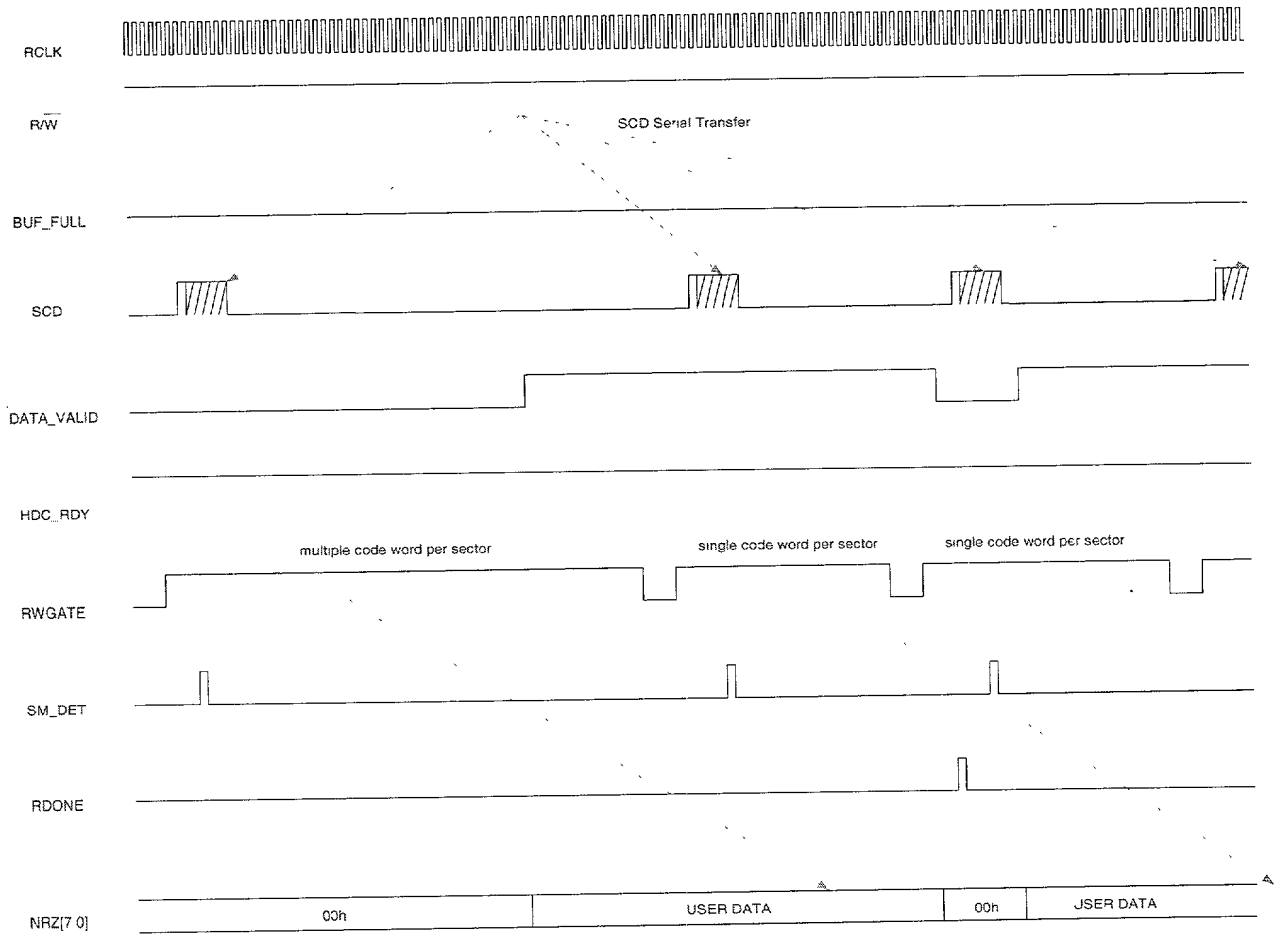

RCLK
R/W
SCD Serial Transfer
BUF_FULL
SCD
DATA_VALID
HDC_RDY
multiple code word per sector
single code word per sector
single code word per sector
RWGATE
SM_DET
RDONE
NRZ[7 0]
00h
USER DATA
00h
JSER DATA

Fig. 12

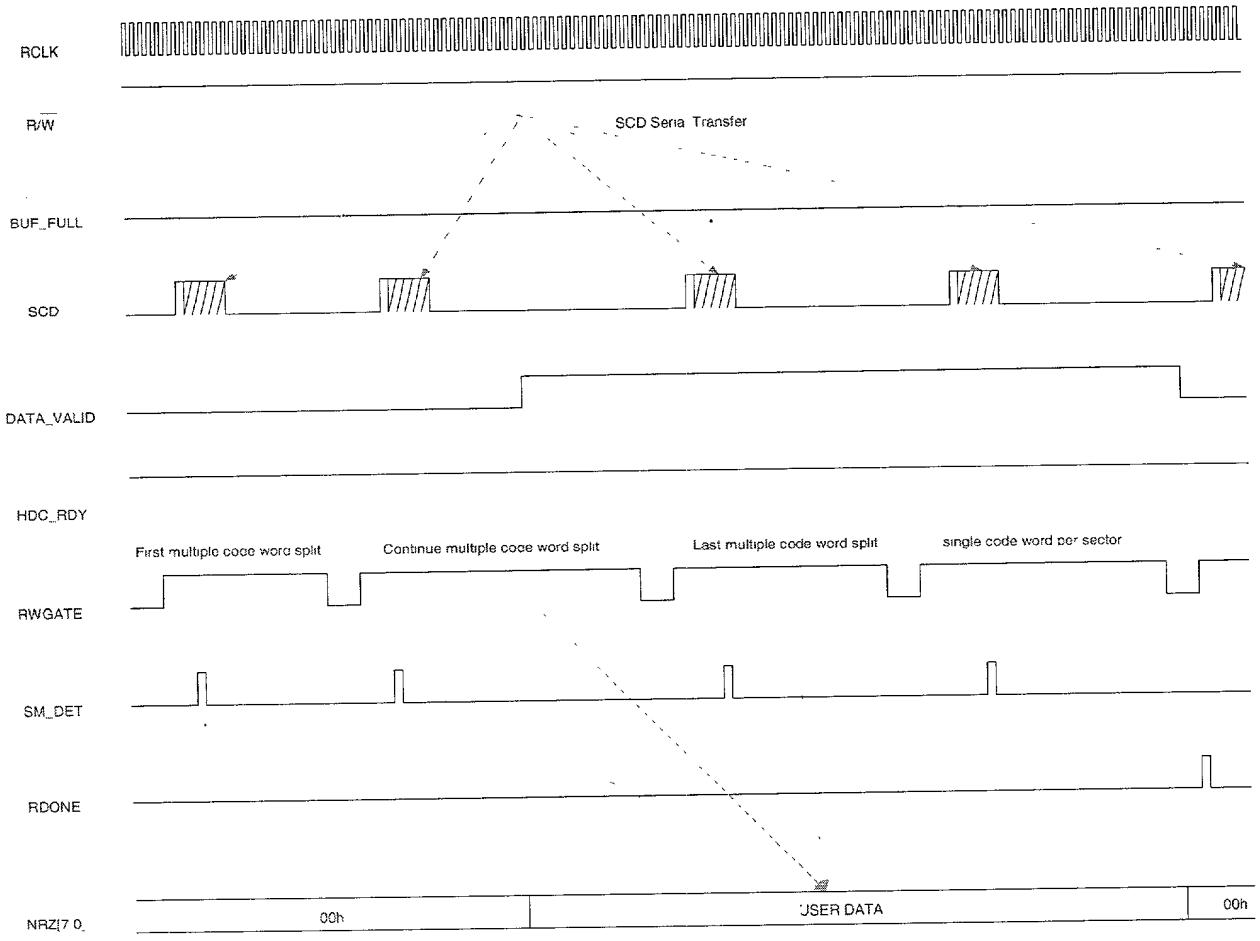

RCLK
R/W
SCD Serial Transfer
BUF_FULL
SCD
DATA_VALID
HDC_RDY
First multiple code word split
Continue multiple code word split
Last multiple code word split
single code word per sector
RWGATE
SM_DET
RDONE
NRZ[7:0]
00h
USER DATA
00h

Fig. 13

20'

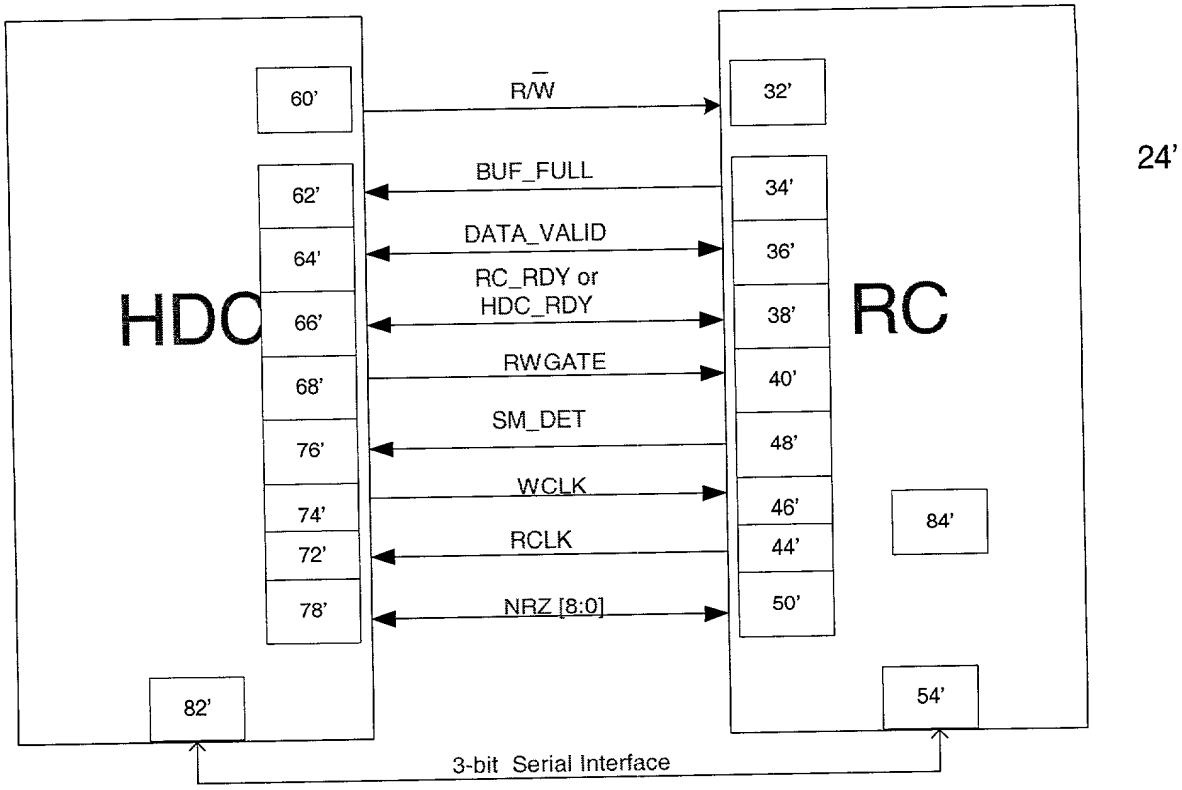

22'
24'
HDC
RC
60'
62'
64'
66'
68'
76'
74'
72'
78'
82'
32'
34'
36'
38'
40'
48'
46'
44'
50'
84'
54'
R/W
BUF_FULL
DATA_VALID
RC_RDY or
HDC_RDY
RWGATE
SM_DET
WCLK
RCLK
NRZ [8:0]
3-bit Serial Interface

Fig. 14

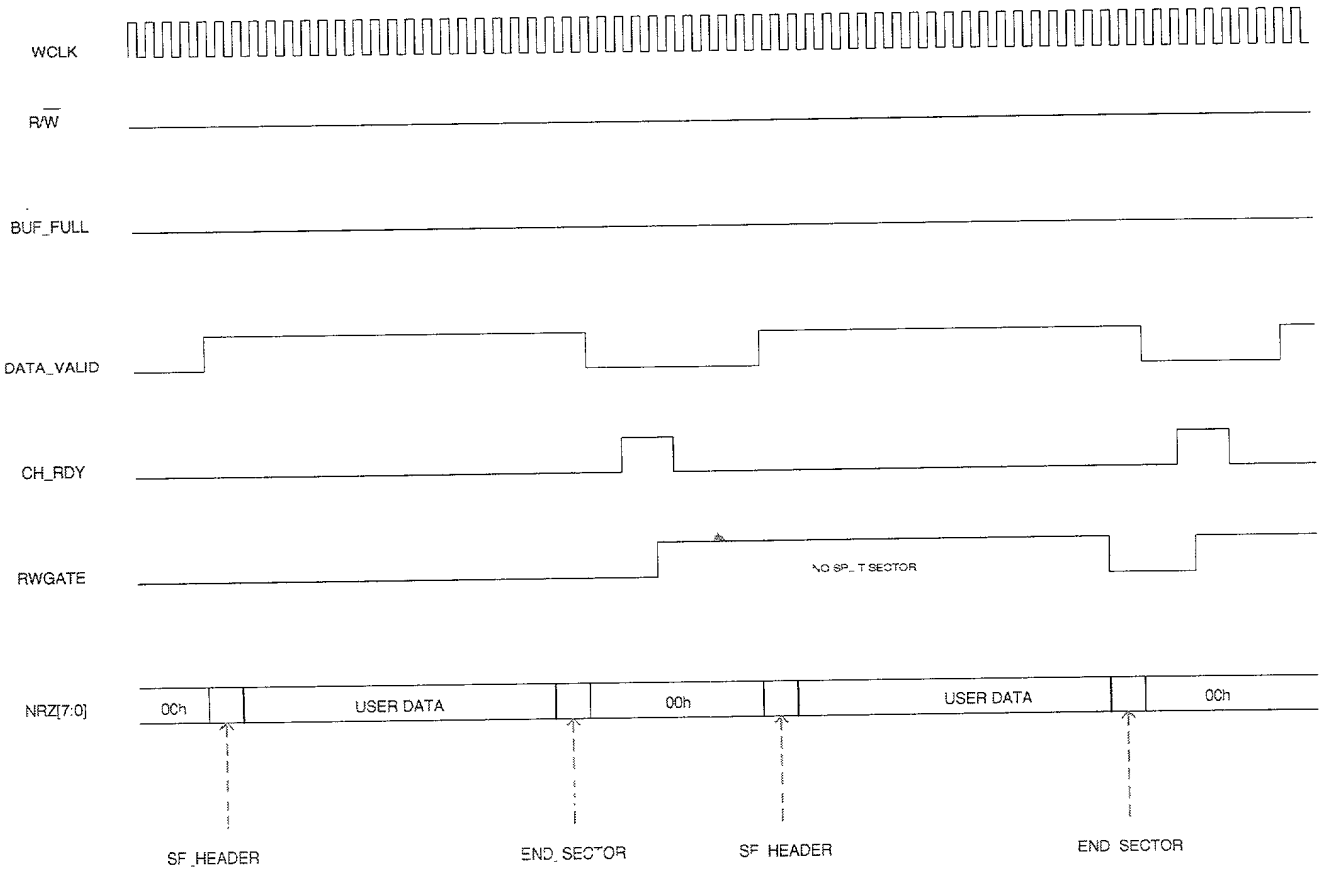

WCLK
R/W
BUF_FULL
DATA_VALID
CH_RDY
RWGATE
NRZ[7:0]
00h
USER DATA
00h
USER DATA
00h
SF_HEADER
END_SECTOR
SF_HEADER
END_SECTOR

Fig. 15

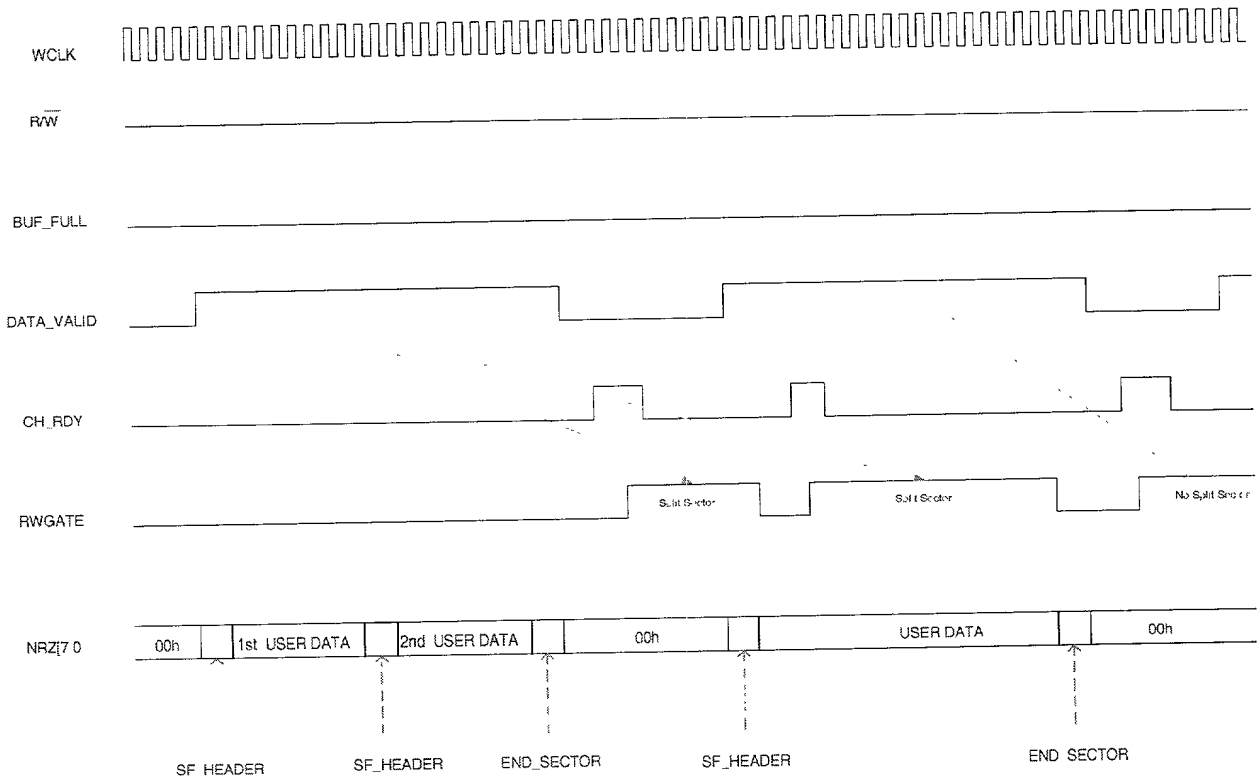

WCLK
R/W
BUF_FULL
DATA_VALID
CH_RDY
RWGATE
NRZ[7:0]
00h
1st USER DATA
2nd USER DATA
00h
USER DATA
00h
SF_HEADER
SF_HEADER
END_SECTOR
SF_HEADER
END SECTOR

Fig. 16

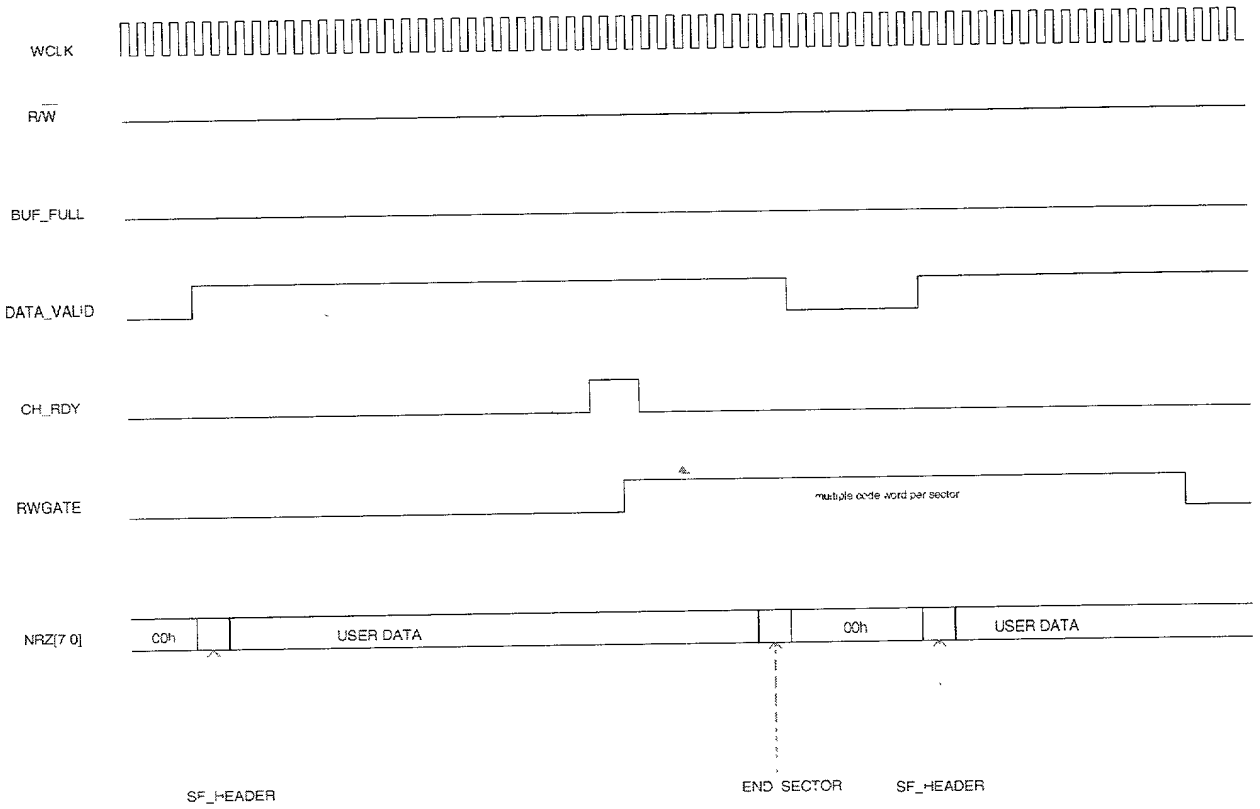

WCLK
R/W
BUF_FULL
DATA_VALID
CH_RDY
RWGATE
multiple code word per sector
NRZ[7:0]
00h
USER DATA
00h
USER DATA
SF_HEADER
END SECTOR
SF_HEADER

Fig. 17

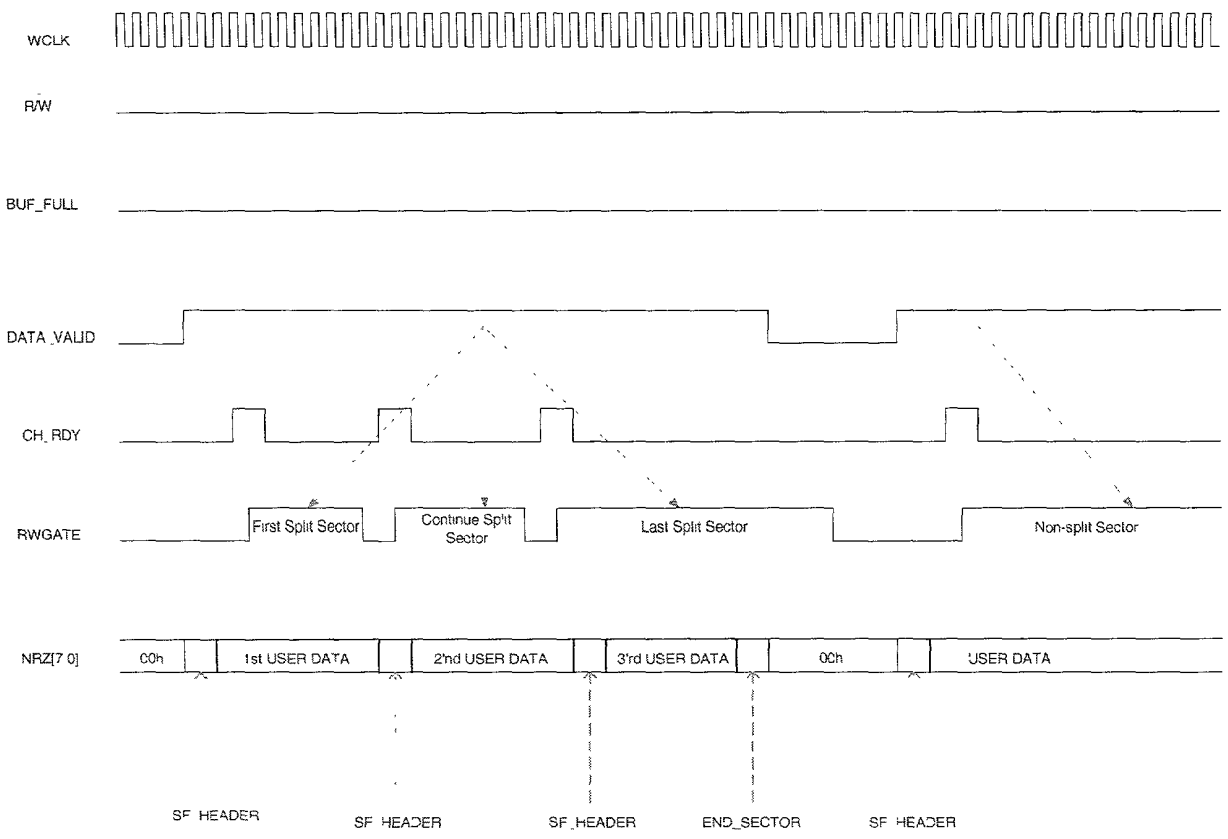

WCLK
R/W
BUF_FULL
DATA_VALID
CH_RDY
RWGATE
First Split Sector
Continue Split Sector
Last Split Sector
Non-split Sector
NRZ[7:0]
C0h
1st USER DATA
2'nd USER DATA
3'rd USER DATA
00h
USER DATA
SF_HEADER
SF_HEADER
SF_HEADER
END_SECTOR
SF_HEADER

Fig. 18

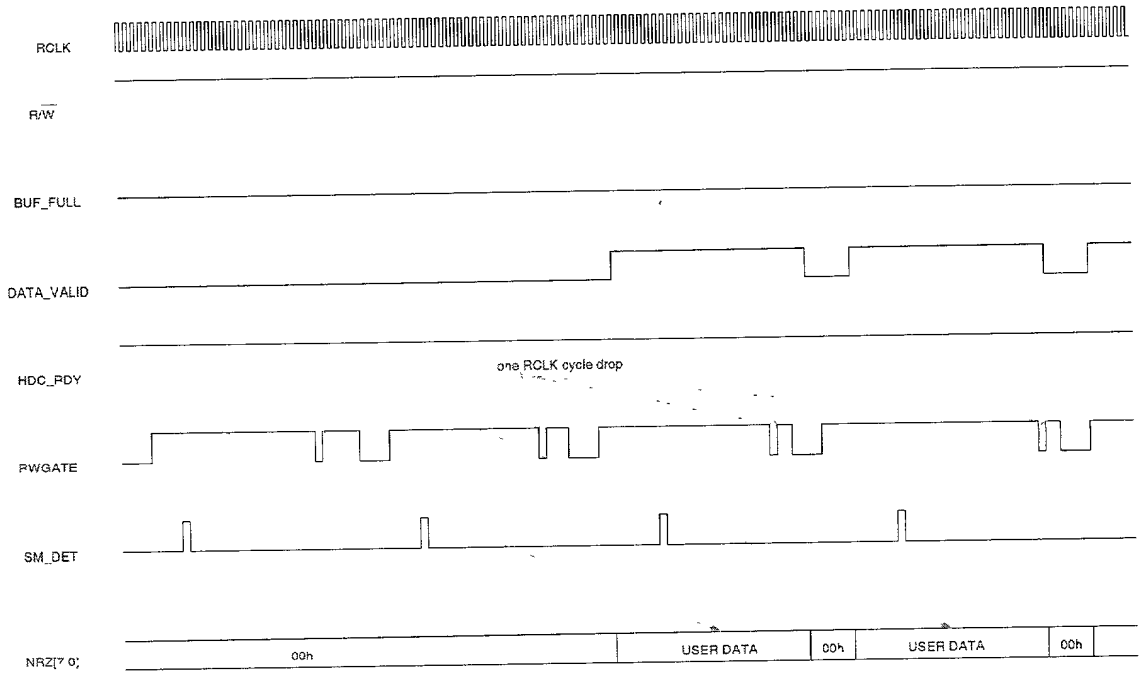

RCLK
R/W
BUF_FULL
DATA_VALID
HDC_RDY
one RCLK cycle drop
RWGATE
SM_DET
NRZ[7:0]
00h
USER DATA
00h
USER DATA
00h

Fig. 19

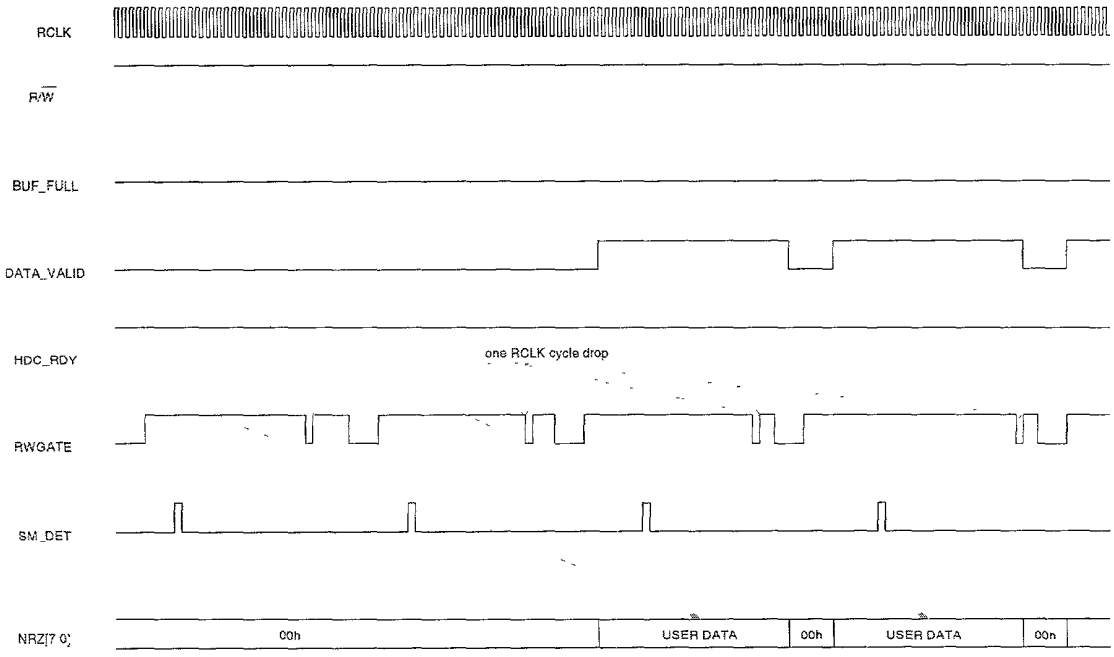

RCLK
R/W
BUF_FULL
DATA_VALID
HDC_RDY
one RCLK cycle drop
RWGATE
SM_DET
NRZ[7:0]
00h
USER DATA
00h
USER DATA
00h

Fig. 20

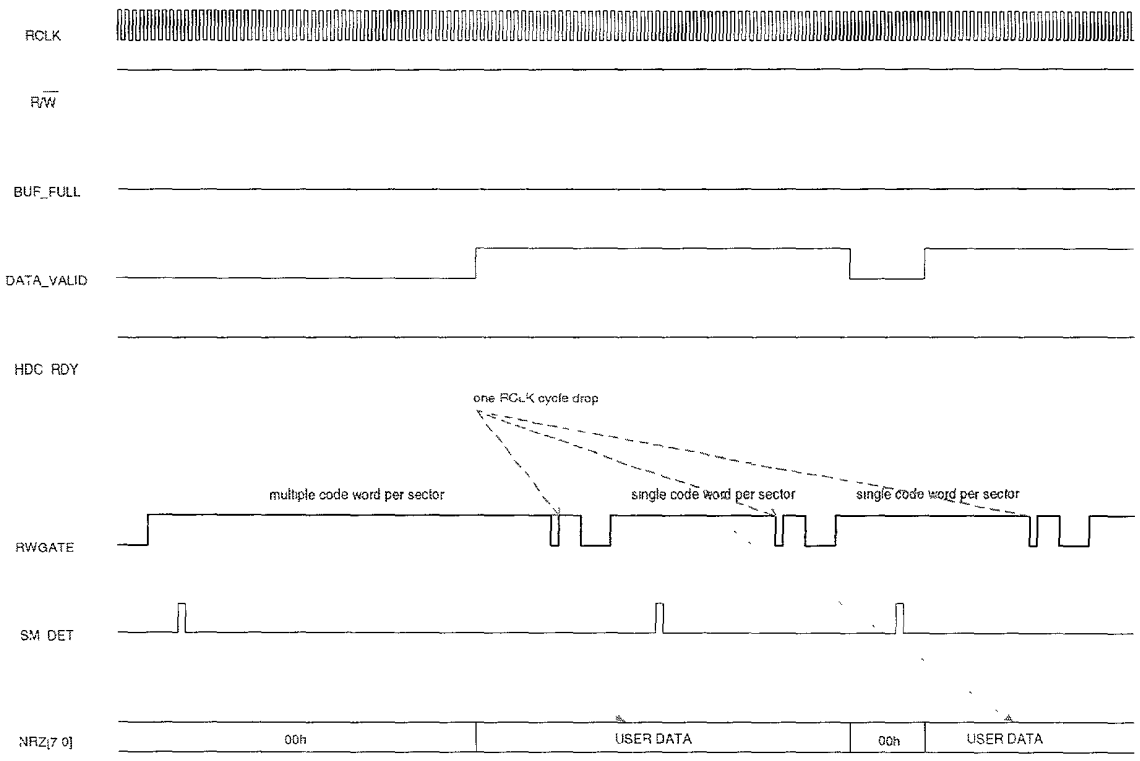

RCLK
R/W
BUF_FULL
DATA_VALID
HDC RDY
one RCLK cycle drop
multiple code word per sector
single code word per sector
single code word per sector
RWGATE
SM DET
NRZ[7:0]
00h
USER DATA
00h
USER DATA

Fig. 21

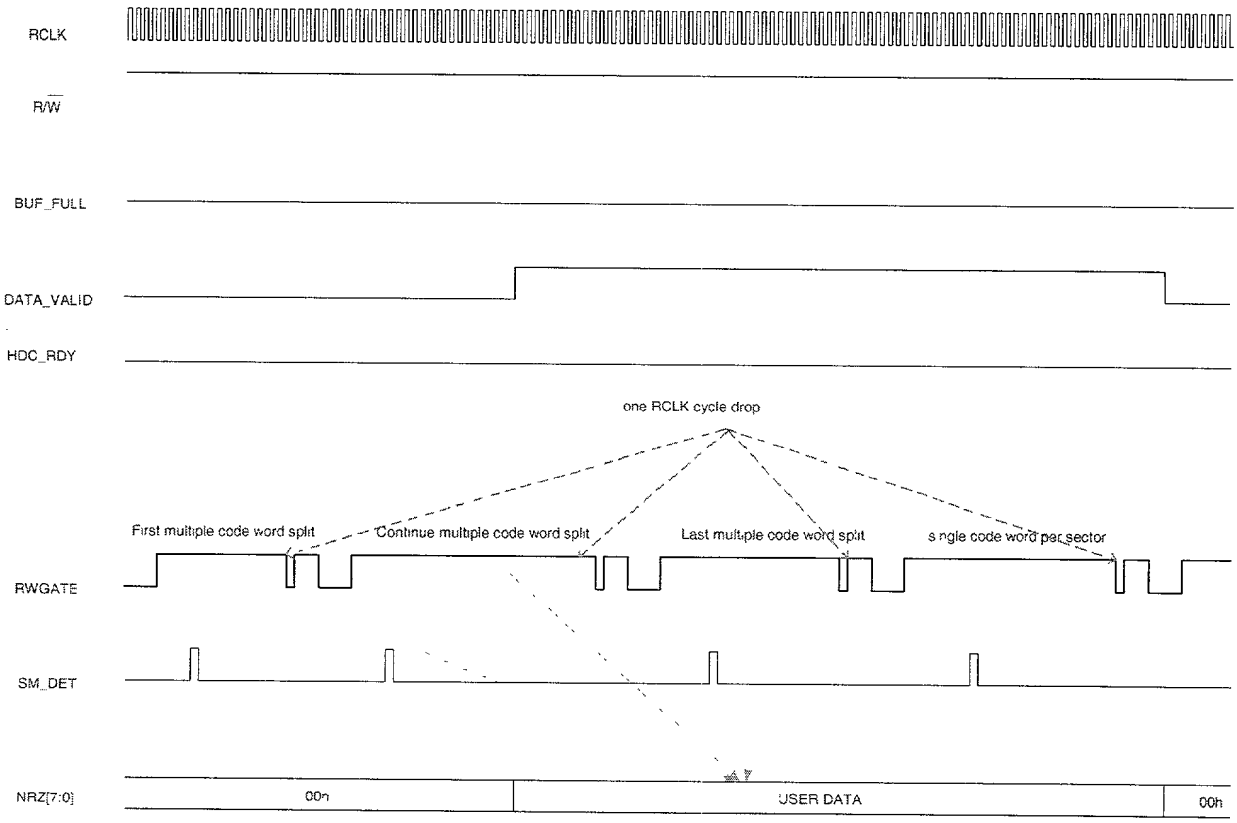

RCLK
R/W
BUF_FULL
DATA_VALID
HDC_RDY
one RCLK cycle drop
First multiple code word split
Continue multiple code word split
Last multiple code word split
single code word per sector
RWGATE
SM_DET
NRZ[7:0]
00h
USER DATA
00h

Fig. 22

SF_HEADER	User data for Multiple codewords	END_SECTOR

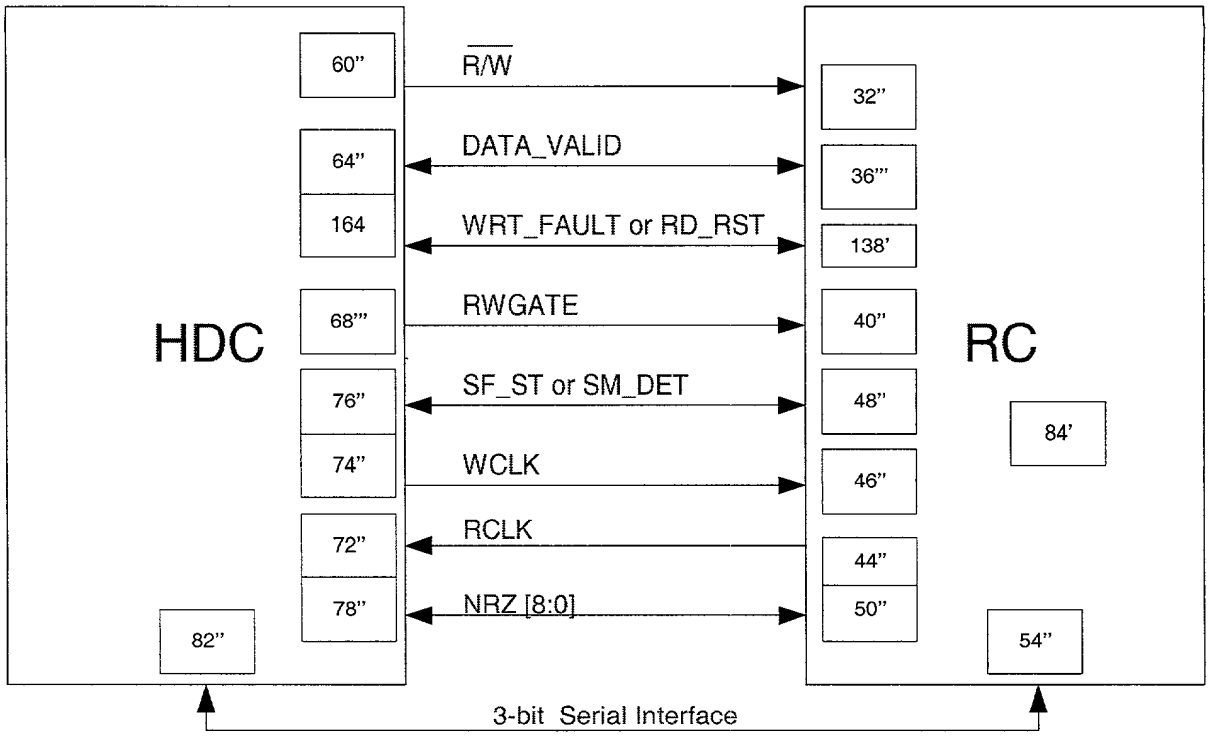
Fig. 23

SF_HEADER	1st set user data	SF_HEADER	2nd set user data	SF_HEADER	3rd set user data	END_SECTOR

Fig. 24

20"

22"



24"

Fig. 25

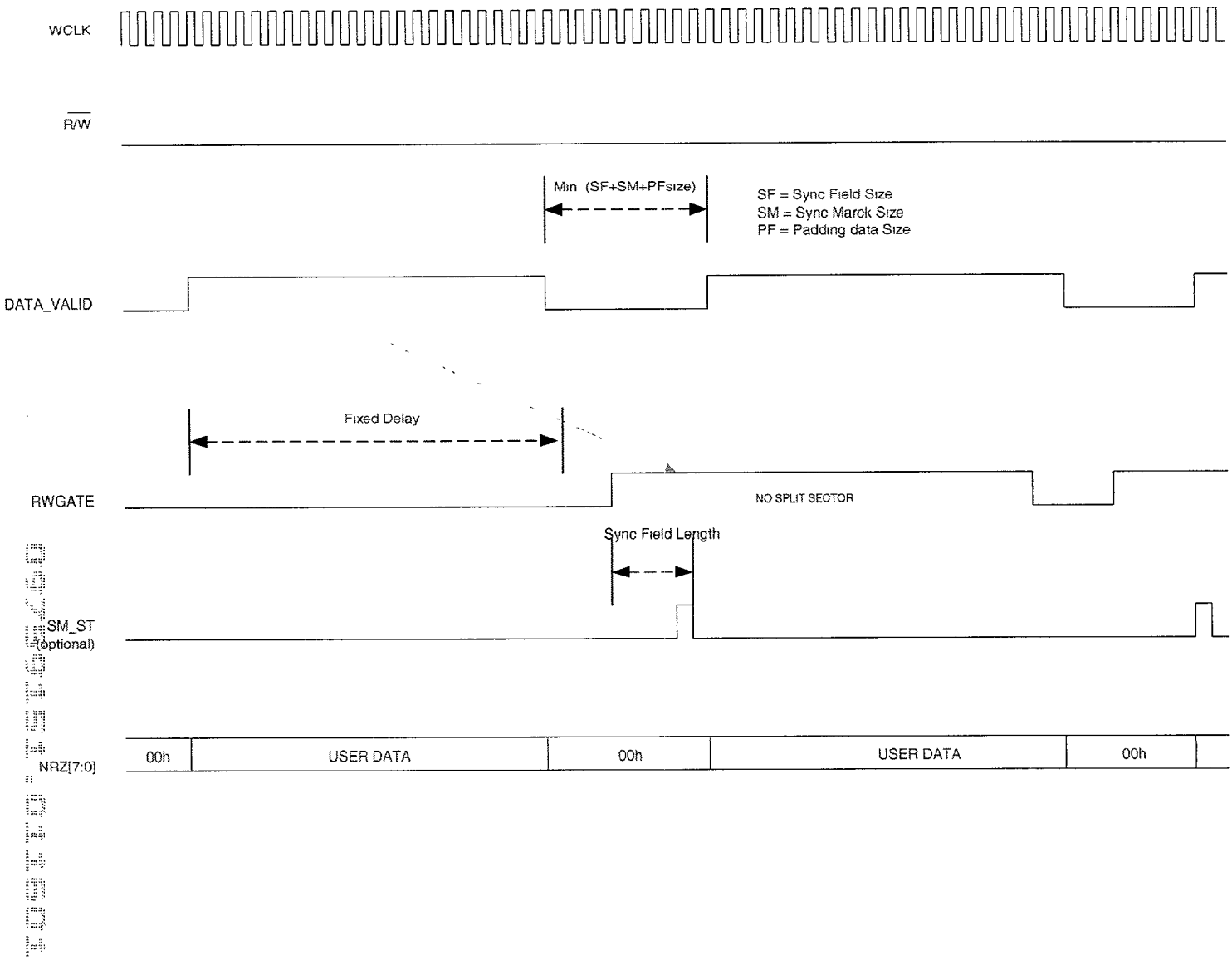

WCLK
R/W
Min (SF+SM+PFsize)
SF = Sync Field Size
SM = Sync Marck Size
PF = Padding data Size
DATA_VALID
Fixed Delay
RWGATE
NO SPLIT SECTOR
Sync Field Length
SM_ST
(optional)
NRZ[7:0]
00h
USER DATA
00h
USER DATA
00h

Fig. 26

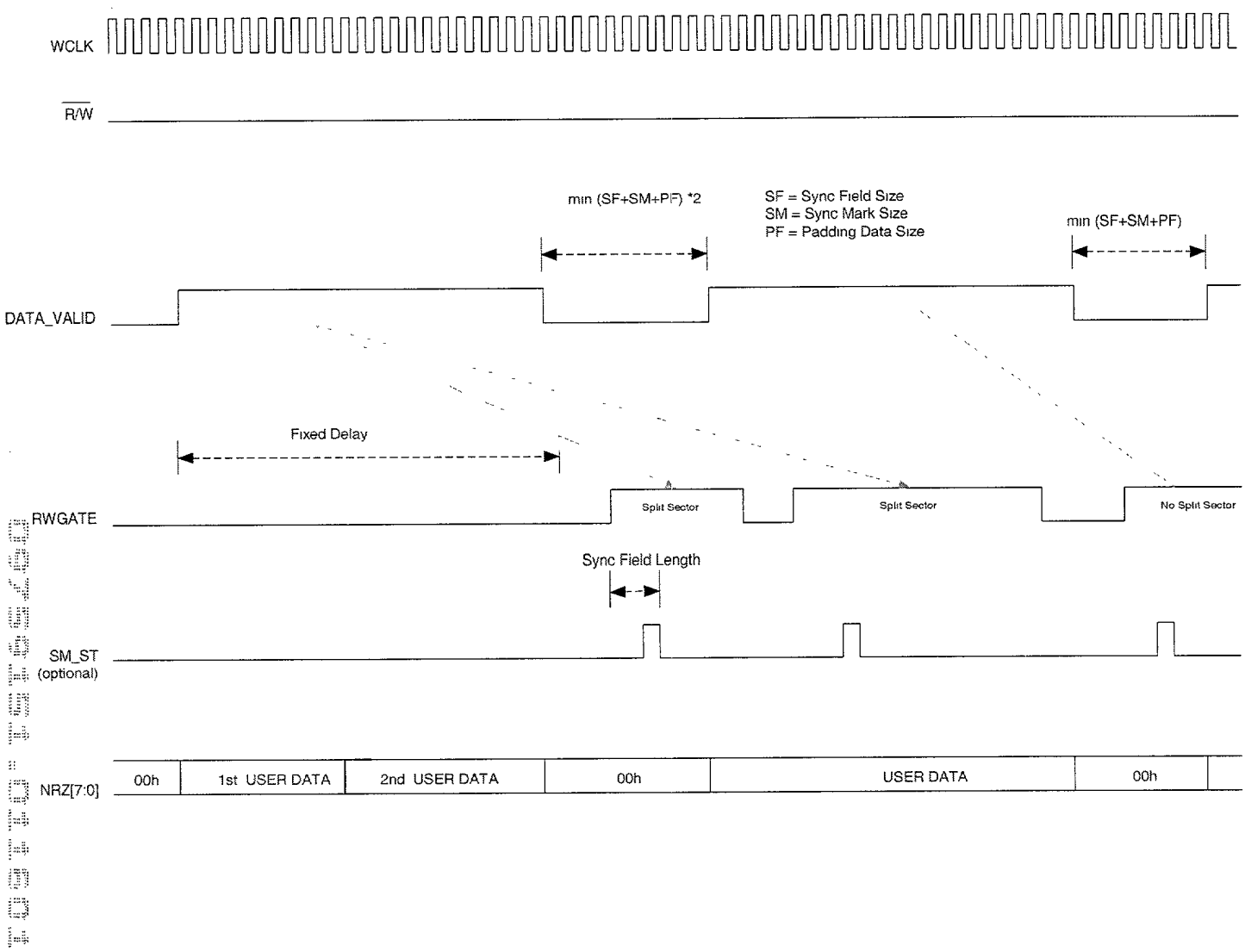

WCLK
R/W
min (SF+SM+PF) *2
SF = Sync Field Size
SM = Sync Mark Size
PF = Padding Data Size
min (SF+SM+PF)
DATA_VALID
Fixed Delay
RWGATE
Split Sector
Split Sector
No Split Sector
Sync Field Length
SM_ST
(optional)
NRZ[7:0]
00h
1st USER DATA
2nd USER DATA
00h
USER DATA
00h

Fig. 27

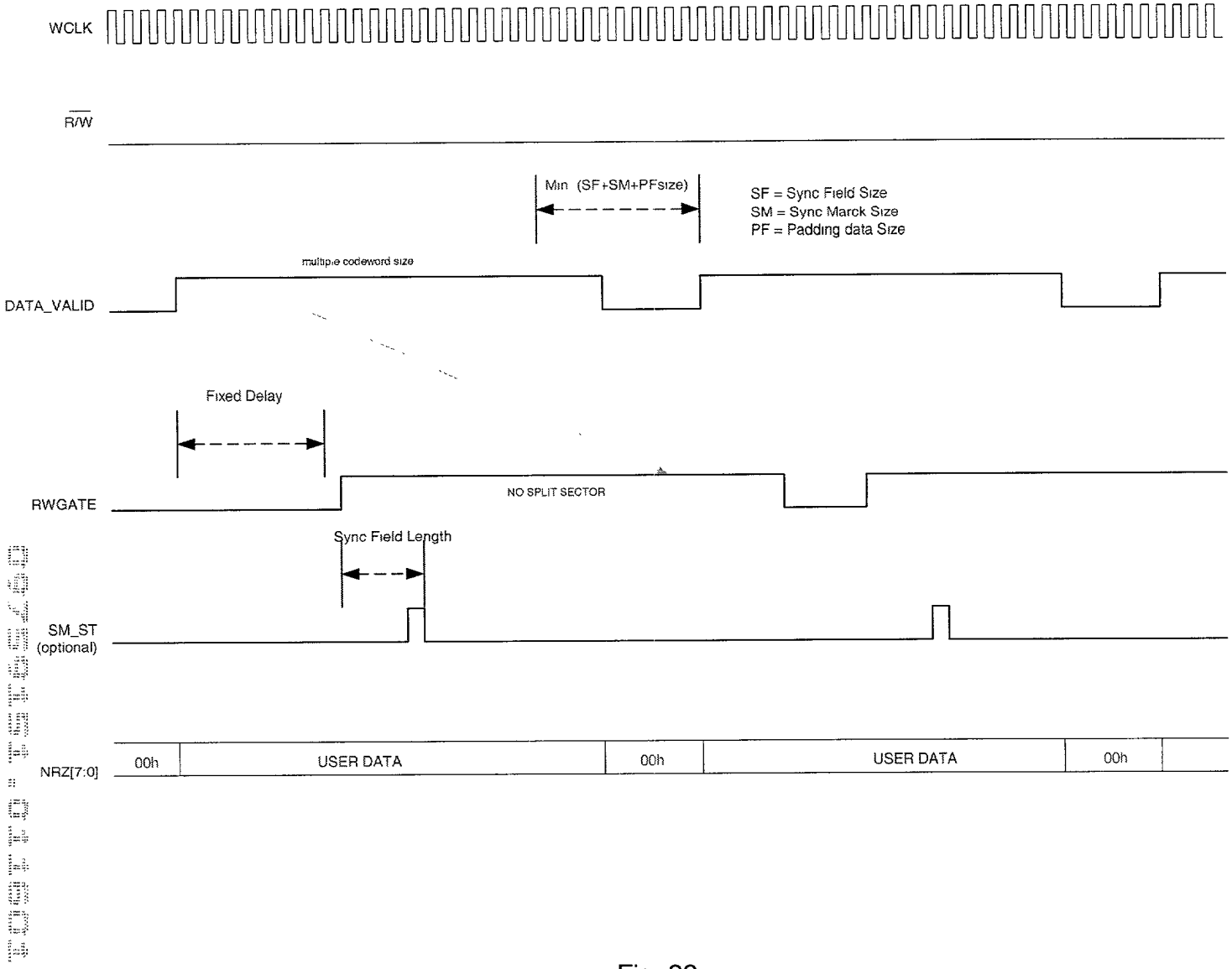

WCLK
R/W
Min (SF+SM+PFsize)
SF = Sync Field Size
SM = Sync Marck Size
PF = Padding data Size
multiple codeword size
DATA_VALID
Fixed Delay
RWGATE
NO SPLIT SECTOR
Sync Field Length
SM_ST
(optional)
NRZ[7:0]
00h
USER DATA
00h
USER DATA
00h

Fig. 28

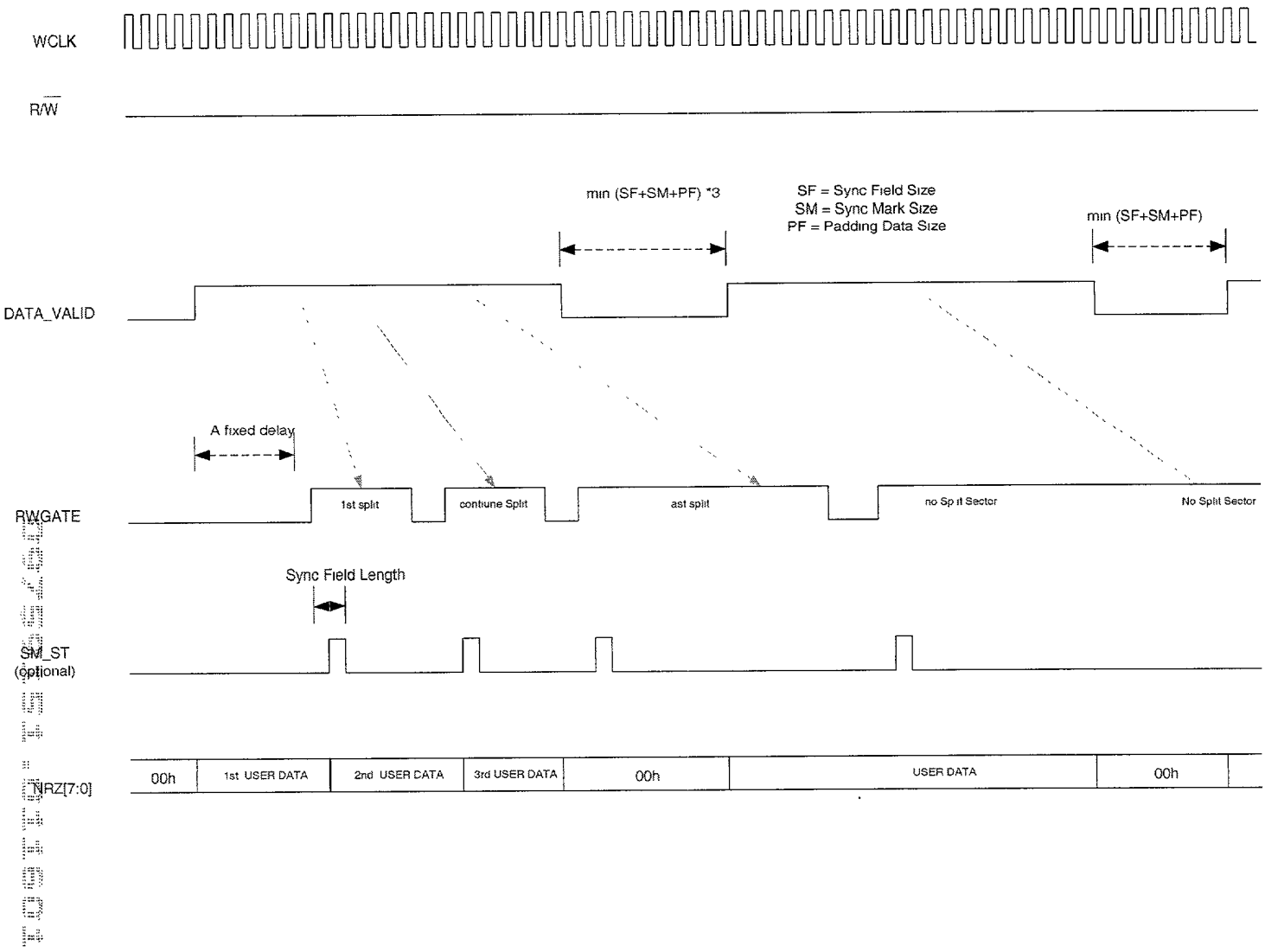

WCLK
R/W
min (SF+SM+PF) *3
SF = Sync Field Size
SM = Sync Mark Size
PF = Padding Data Size
min (SF+SM+PF)
DATA_VALID
A fixed delay
RWGATE
1st split
continue Split
ast split
no Sp it Sector
No Split Sector
Sync Field Length
SM_ST
(optional)
NRZ[7:0]
00h
1st USER DATA
2nd USER DATA
3rd USER DATA
00h
USER DATA
00h

Fig. 29

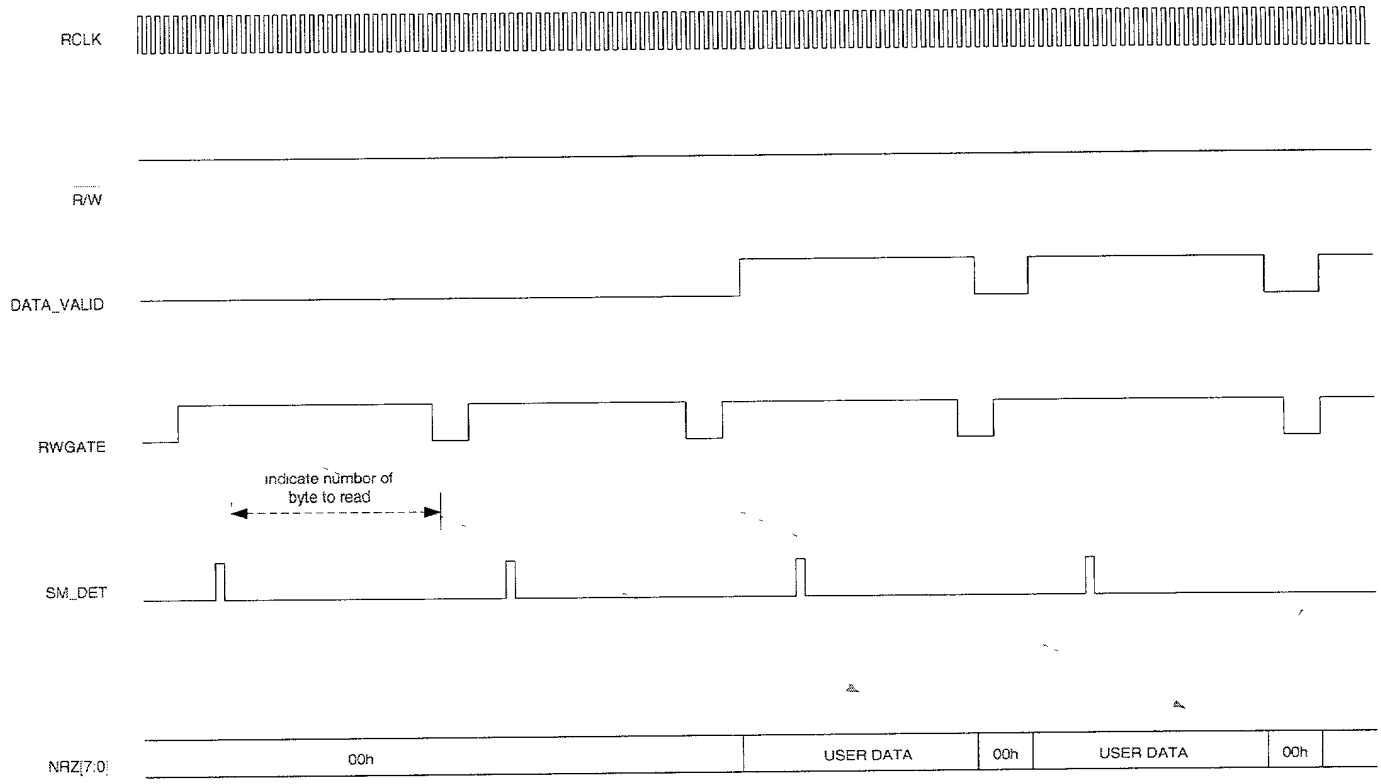

RCLK
R/W
DATA_VALID
RWGATE
indicate number of byte to read
SM_DET
NRZ[7:0]
00h
USER DATA
00h
USER DATA
00h

Fig. 30

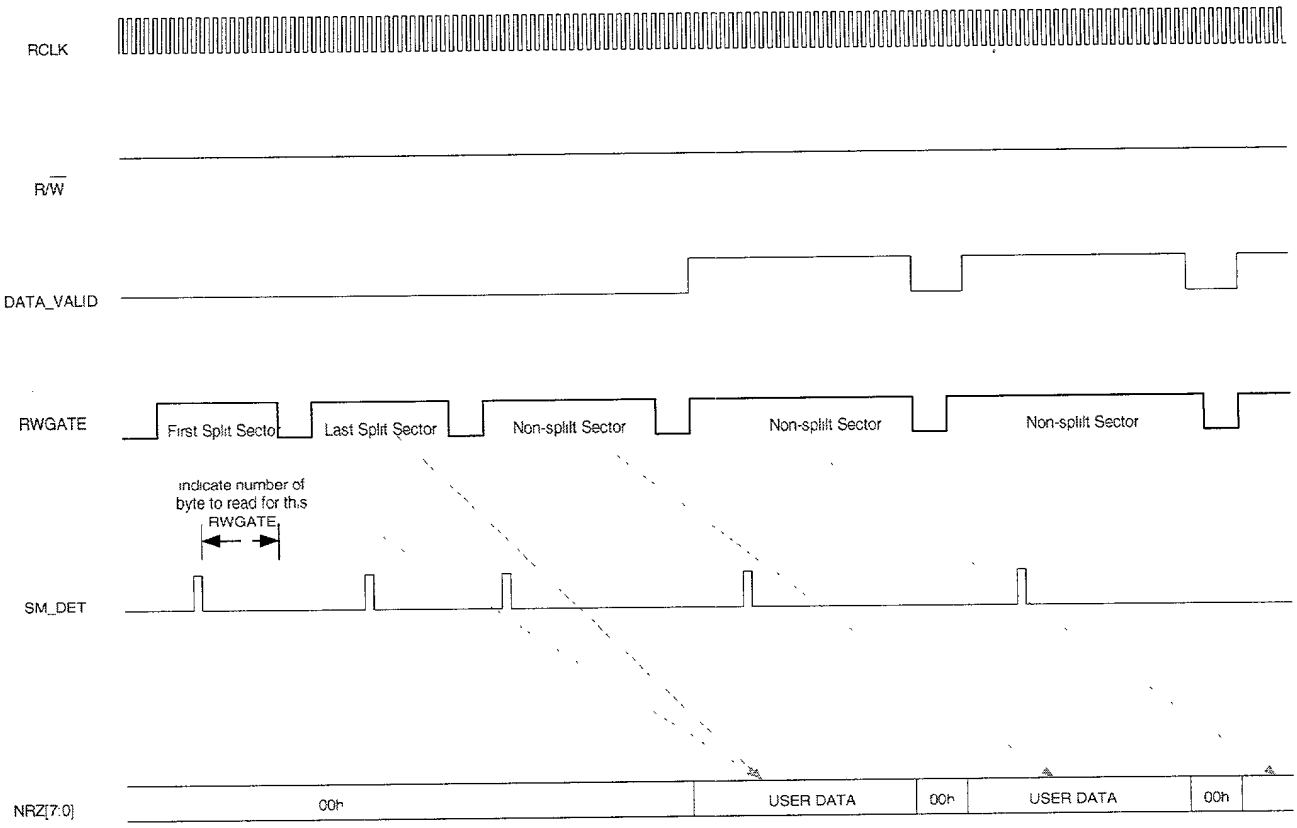

RCLK
R/W
DATA_VALID
RWGATE
First Split Secto
Last Split Sector
Non-split Sector
Non-split Sector
Non-split Sector
indicate number of byte to read for this RWGATE
SM_DET
NRZ[7:0]
00h
USER DATA
00h
USER DATA
00h

Fig. 31

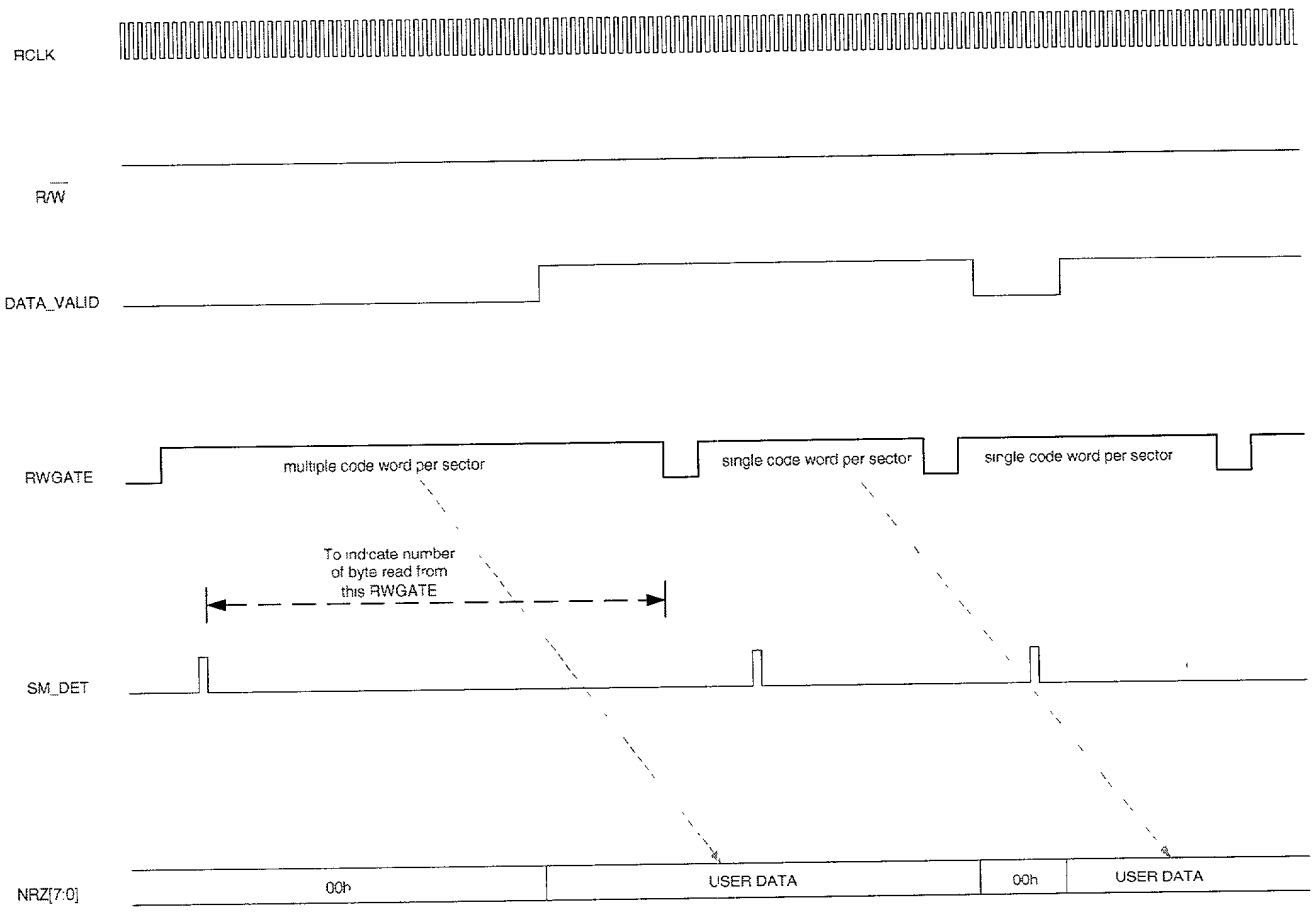

RCLK
R/W
DATA_VALID
RWGATE
multiple code word per sector
single code word per sector
single code word per sector
To indicate number of byte read from this RWGATE
SM_DET
NRZ[7:0]
00h
USER DATA
00h
USER DATA

Fig. 32

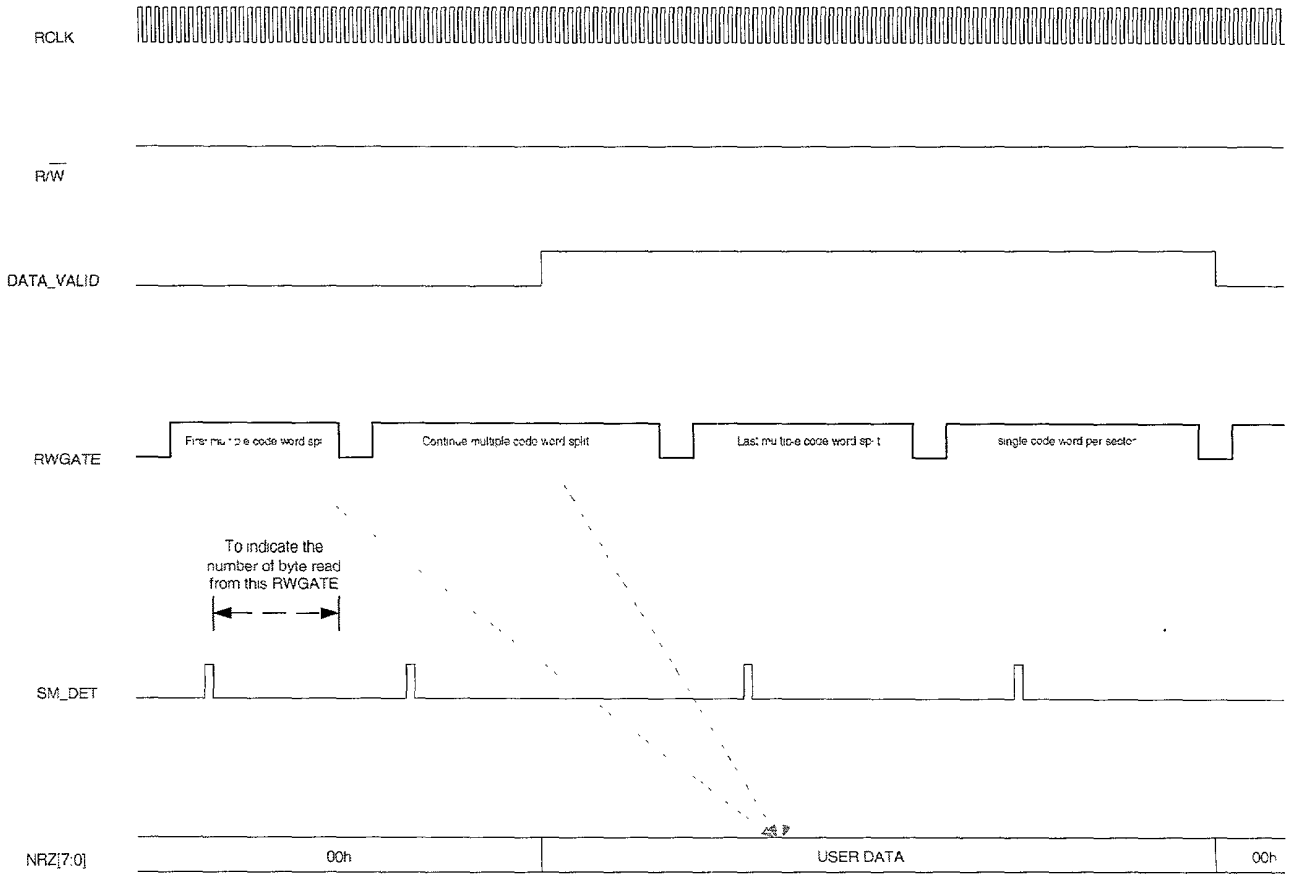

RCLK
R/W
DATA_VALID
RWGATE
Continue multiple code word split
single code word per sector
To indicate the
number of byte read
from this RWGATE
SM_DET
NRZ[7:0]
00h
USER DATA
00h

Fig. 33

20'''

22'''

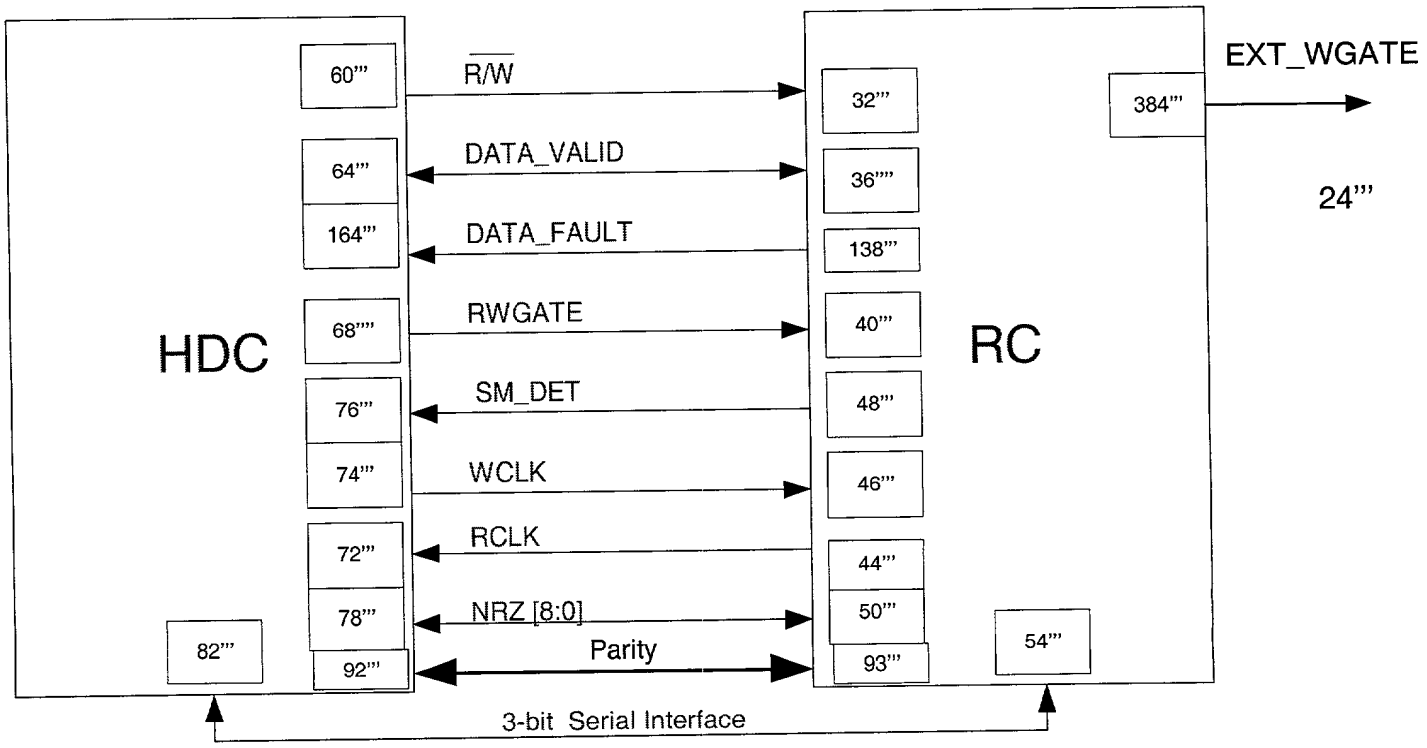

HDC
RC
60'''
64'''
164'''
68''''
76'''
74'''
72'''
78'''
82'''
92'''
R/W
DATA_VALID
DATA_FAULT
RWGATE
SM_DET
WCLK
RCLK
NRZ [8:0]
Parity
3-bit Serial Interface
32'''
36''''
138'''
40'''
48'''
46'''
44'''
50'''
93'''
54'''
384'''
EXT_WGATE
24'''

Fig. 34

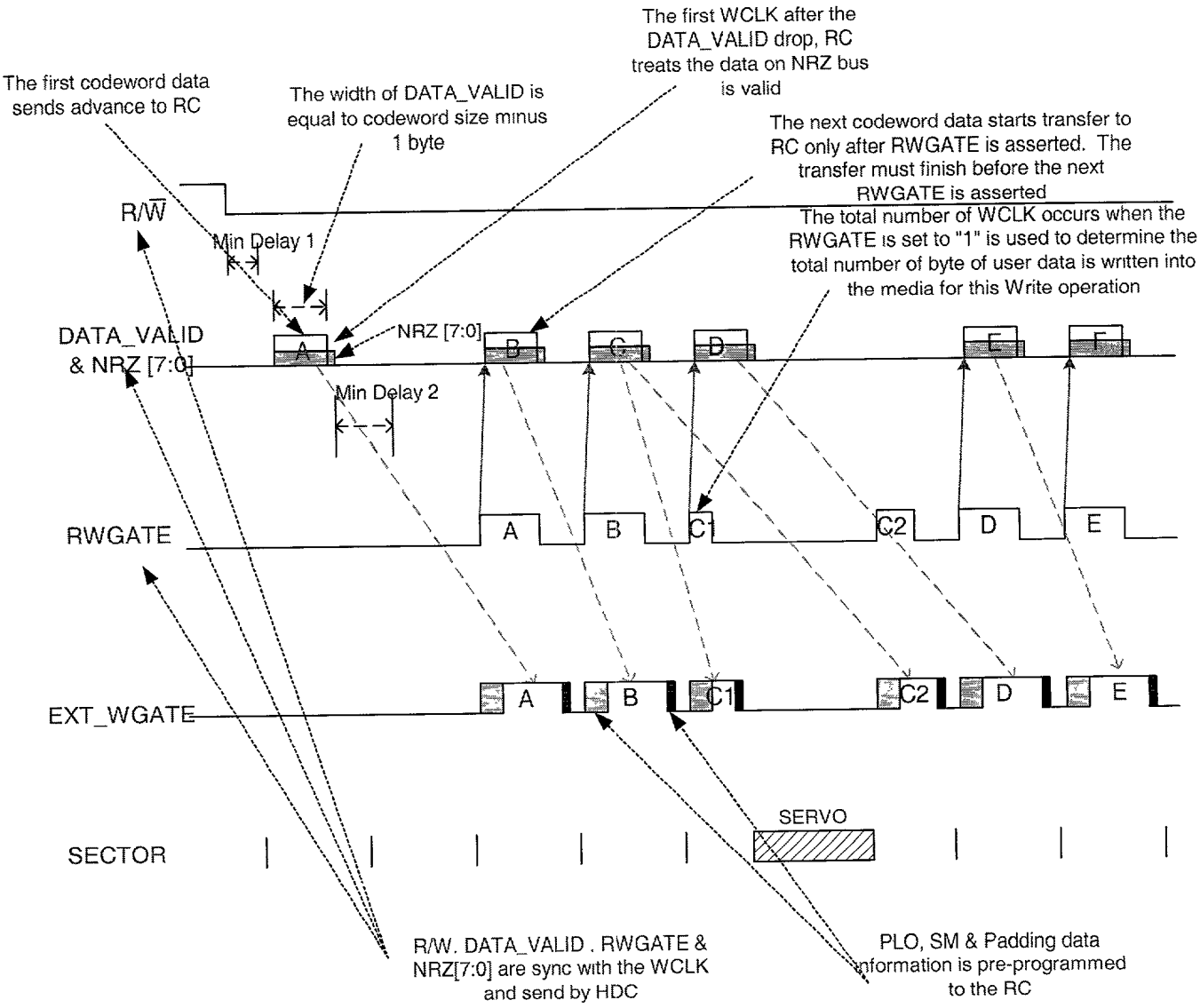

The first codeword data sends advance to RC
The width of DATA_VALID is equal to codeword size minus 1 byte
The first WCLK after the DATA_VALID drop, RC treats the data on NRZ bus is valid
The next codeword data starts transfer to RC only after RWGATE is asserted. The transfer must finish before the next RWGATE is asserted
The total number of WCLK occurs when the RWGATE is set to "1" is used to determine the total number of byte of user data is written into the media for this Write operation
R/W
Min Delay 1
DATA_VALID & NRZ [7:0]
NRZ [7:0]
Min Delay 2
RWGATE
EXT_WGATE
SECTOR
SERVO
A
B
C1
C2
D
E
F
R/W. DATA_VALID . RWGATE & NRZ[7:0] are sync with the WCLK and send by HDC
PLO, SM & Padding data information is pre-programmed to the RC

Fig. 35

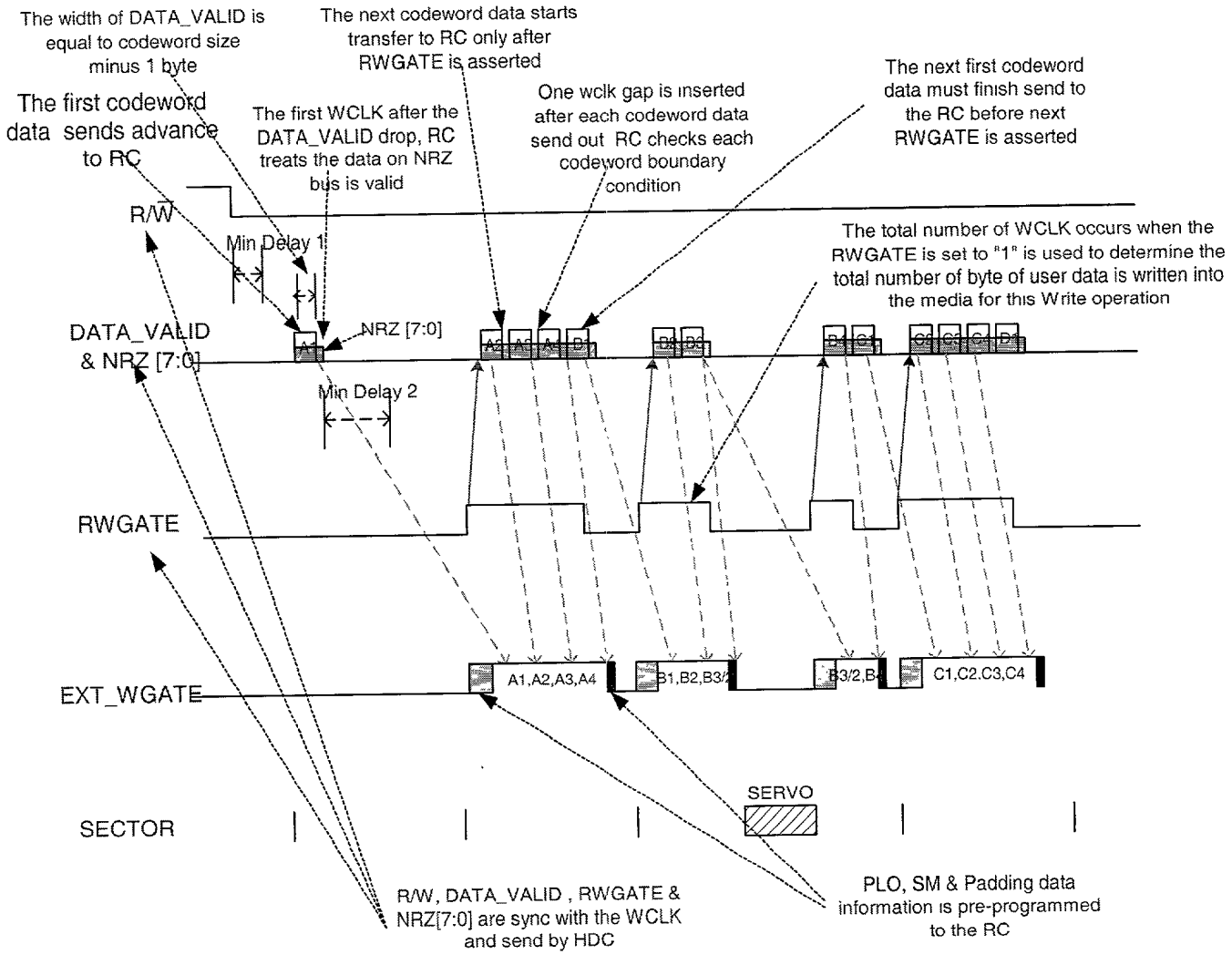

The width of DATA_VALID is equal to codeword size minus 1 byte
The next codeword data starts transfer to RC only after RWGATE is asserted
The first codeword data sends advance to RC
The first WCLK after the DATA_VALID drop, RC treats the data on NRZ bus is valid
One wclk gap is inserted after each codeword data send out RC checks each codeword boundary condition
The next first codeword data must finish send to the RC before next RWGATE is asserted
R/W
Min Delay 1
The total number of WCLK occurs when the RWGATE is set to "1" is used to determine the total number of byte of user data is written into the media for this Write operation
DATA_VALID & NRZ [7:0]
NRZ [7:0]
Min Delay 2
RWGATE
EXT_WGATE
A1,A2,A3,A4
B1,B2,B3/2
B3/2,B4
C1,C2,C3,C4
SERVO
SECTOR
R/W, DATA_VALID , RWGATE & NRZ[7:0] are sync with the WCLK and send by HDC
PLO, SM & Padding data information is pre-programmed to the RC

Fig. 36

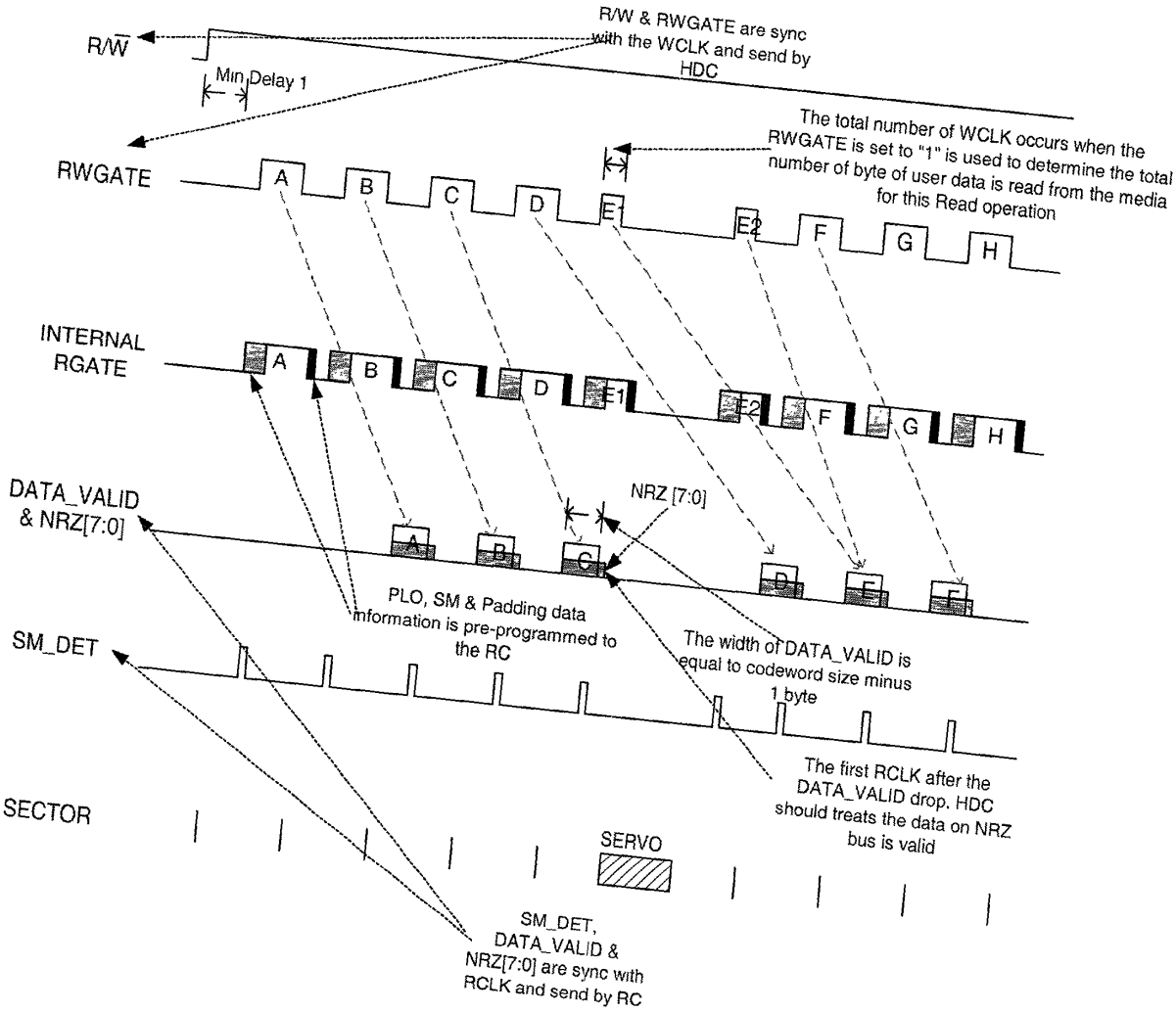

R/W
R/W & RWGATE are sync with the WCLK and send by HDC
Min Delay 1
RWGATE
The total number of WCLK occurs when the RWGATE is set to "1" is used to determine the total number of byte of user data is read from the media for this Read operation
A
B
C
D
E1
E2
F
G
H
INTERNAL RGATE
DATA_VALID & NRZ[7:0]
NRZ [7:0]
PLO, SM & Padding data information is pre-programmed to the RC
The width of DATA_VALID is equal to codeword size minus 1 byte
SM_DET
The first RCLK after the DATA_VALID drop. HDC should treats the data on NRZ bus is valid
SECTOR
SERVO
SM_DET, DATA_VALID & NRZ[7:0] are sync with RCLK and send by RC

Fig. 37

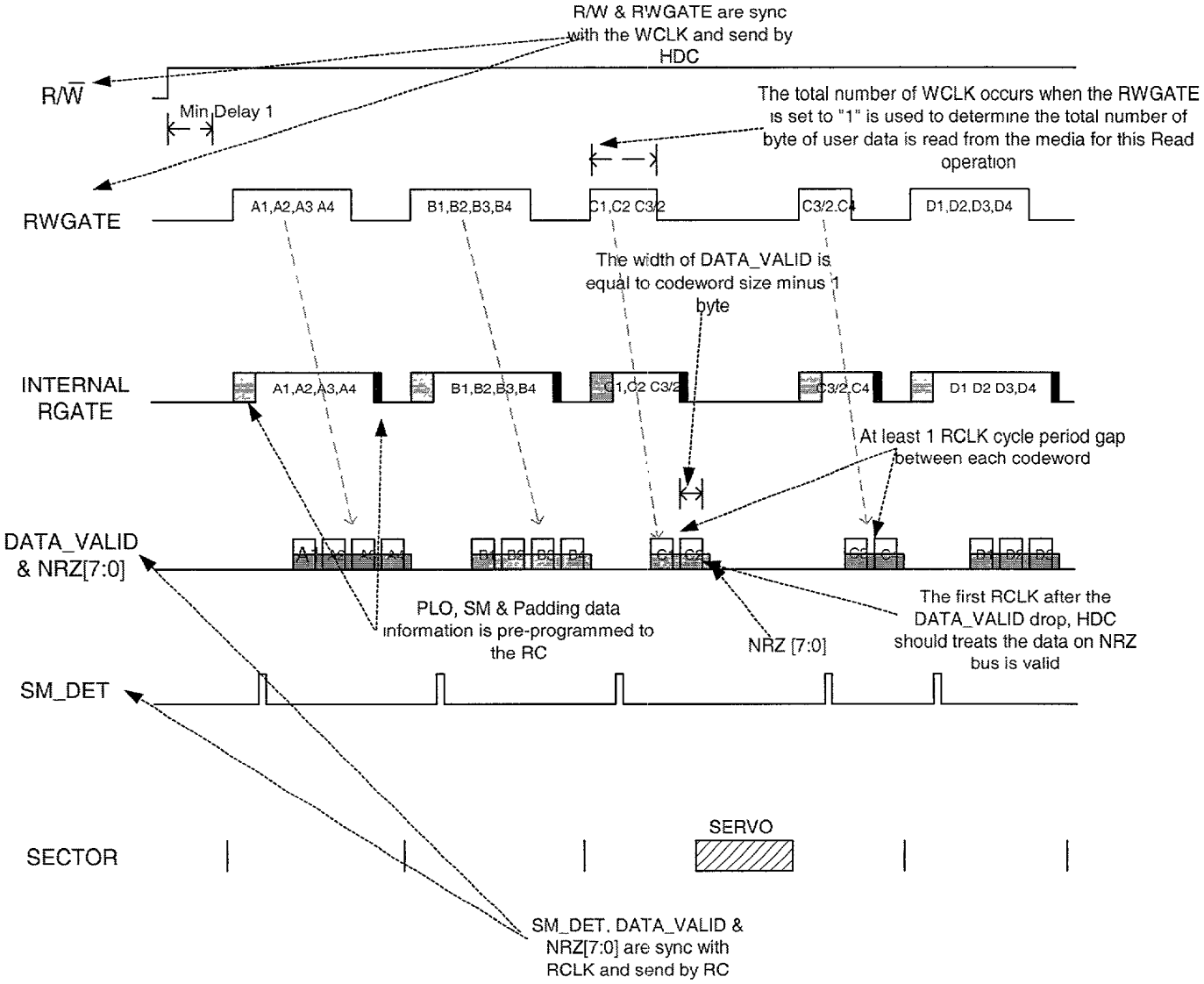

R/W & RWGATE are sync with the WCLK and send by HDC
R/W
Min Delay 1
The total number of WCLK occurs when the RWGATE is set to "1" is used to determine the total number of byte of user data is read from the media for this Read operation
RWGATE
A1,A2,A3 A4
B1,B2,B3,B4
C1,C2 C3/2
C3/2,C4
D1,D2,D3,D4
The width of DATA_VALID is equal to codeword size minus 1 byte
INTERNAL RGATE
A1,A2,A3,A4
B1,B2,B3,B4
C1,C2 C3/2
C3/2,C4
D1 D2 D3,D4
At least 1 RCLK cycle period gap between each codeword
DATA_VALID & NRZ[7:0]
PLO, SM & Padding data information is pre-programmed to the RC
NRZ [7:0]
The first RCLK after the DATA_VALID drop, HDC should treats the data on NRZ bus is valid
SM_DET
SERVO
SECTOR
SM_DET, DATA_VALID & NRZ[7:0] are sync with RCLK and send by RC

Fig. 38